THE KEY HANDOFF: RETRIEVAL IN HYBRID LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

A two-hop question makes a language model retrieve twice: once to produce a bridge entity, and once to retrieve with it. Transformers resolve that entity in their early layers. Hybrid models replace most of the attention with a recurrent state, so where the key becomes usable, and where it is spent on an answer, is not known. Answering "Where is the ball?" from "the ball belongs to Alice" and "Alice is in the garden" turns on Alice, a name the question does not mention. A model could reach garden through Alice, the key it computed, or through where the fact sits in the prompt. Across twelve models, dense and hybrid, we move a hidden state from one story into another where the two routes lead to different places, and read off which place the model gives. An attention layer converts the key in every model we tested, and crossing that layer removes its usable effect, all of it where no attention follows. In sequential hybrids this makes the answer a handoff: recurrent layers carry the key forward, and attention spends it. Retrieval does not always stop there. Writing a different fact into the memory of the recurrent layers that follow the last attention layer can move the answer toward that fact, multiplying its odds by **1.3** to **2.7**, and which hybrids do this is not settled by their architecture or training. That read is addressed by the key, not by position: recurrent state in a hybrid is not only a carrier, but a memory that later layers can query.

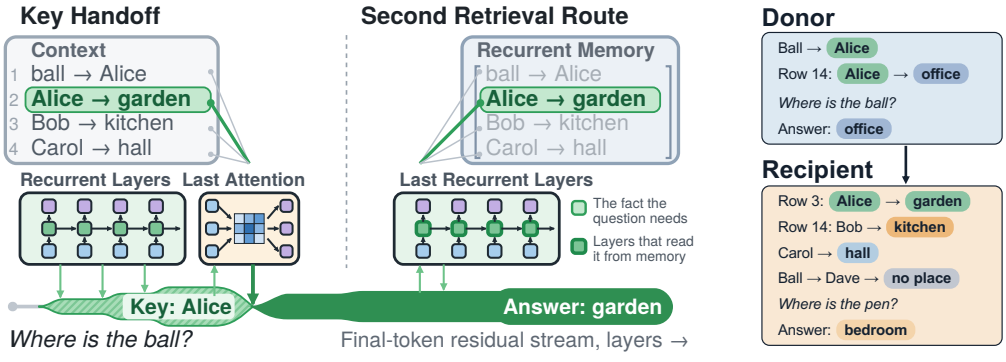
1 INTRODUCTION

A ball belongs to Alice, and Alice is in the garden. To answer *Where is the ball?*, a language model first resolves Alice, then uses Alice as a *retrieval key* to find the garden. We ask where a usable key emerges, where it becomes an answer, and which components perform those operations.

Hybrid language models interleave attention, which accesses earlier token representations, with recurrent computation, which maintains a state as it reads the sequence (Gu & Dao, 2024; Dao & Gu, 2024; Waleffe et al., 2024; NVIDIA et al., 2025), whereas dense transformers apply attention at every layer, so the same two-hop task can be solved with the intermediate computation distributed differently. For transformers this computation has a known shape: the intermediate entity is resolved in the early layers (Biran et al., 2024; Yang et al., 2024), and a small set of attention heads does the retrieving (Wu et al., 2025). Hybrids remove most of that attention. That is the point of the design: a recurrent layer holds one state instead of every earlier token. But attention is what prior work ties retrieval to, and a hybrid has the least of it. Work on them so far finds retrieval still leaning on the attention that remains (Michalak & Abreu, 2025; Bick et al., 2025). What the recurrent layers contribute, and whether any of the work happens where no attention is left to do it, is open.

Our **crossed assay** distinguishes a computed key from a carried answer or a pointer to a fact's position by moving a state from one story (the *donor*) into another (the *recipient*) whose facts are arranged so that the three predict three different answers (Figure 1b). Building on counterfactual binding studies (Feng & Steinhardt, 2024; Gur-Arieh et al., 2026), it varies the key and the position of the donor's fact independently, and scores both against an answer no account predicts. The key turns out to move through a sequential hybrid as a handoff: recurrent layers carry it, an attention layer spends it, and retrieval does not always stop there (Figure 1a). We reach that picture with one measurement and three results:

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) (b)

Figure 1: **The handoff, and a pair that measures it.** (a) In a sequential hybrid the final token's stream is hatched while it carries the key to an attention layer that reads it, and solid once it carries the answer; after the last attention layer a recurrent layer can retrieve a binding from its own memory. (b) The two prompts share a row, so the donor's carried answer, that row, and the key it computes point at three different places; hall is predicted by none and serves as the reference. Only the donor's final-token state moves, into the same layer input of the recipient. Names and places stand in for the arrow task's keys and numbers; the literal prompts are in Appendix A.

- **A measurement for computed keys.** An activation-patching assay that follows a computed key across architectures and kinds of intervention, over 23 depth profiles from twelve checkpoints in nine model families (Section 2).
- **Where keys become answers.** In every profile the key peaks at the input of the layer that converts it, and crossing that layer removes the key's usable effect: all of it where that layer is the model's last attention layer, part of it where attention follows (Section 3).
- **A division of labor in hybrids.** In four hybrids recurrent layers carry the key to an attention layer, where a few heads resolve it against the facts in context (Section 4).
- **Retrieval after attention.** Writing a different fact into the recurrent memory that follows the last attention layer moves the answer toward it, and that read is addressed by the key rather than by position (Section 5).

2 A CROSSED ASSAY FOR COMPUTED-KEY RETRIEVAL

A donor and a recipient. In Figure 1b the *donor* asks where the ball is: the ball belongs to Alice, and the fact placing her in the office sits at row 14. The *recipient* asks where the pen is and answers bedroom; among its 17 facts it places Alice in the garden, Bob in the kitchen at row 14, and Carol in the hall. The ball's owner has no stated location, so a transplanted query token alone has nothing to resolve here (Appendix B). After the transplant a carried answer predicts office, a computed Alice key predicts garden (*key-based* retrieval), and a row-14 pointer predicts kitchen (*row-based* retrieval, positional retrieval in the sense of Gur-Arieh et al. (2026)). Carol's hall is predicted by none of the three and serves as the *reference candidate*.

The transplant. For donor x_d and recipient x_r, let h_l be the *residual state* entering layer l, the representation passed between layers. At the final prompt token we copy $h_l(x_d)$ into the recipient at the same layer input, and the recipient finishes its forward pass with its own tokens and its computation over earlier tokens unchanged. No learned transformation is involved (Geiger et al., 2021). For each candidate answer c we measure the change in its logit ℓ against the same recipient without intervention:

$$h_l(x_r) \leftarrow h_l(x_d), \qquad \Delta\ell(c) = \ell\big(c \mid x_r, h_l(x_d)\big) - \ell(c \mid x_r). \tag{1}$$

A positive change promotes that candidate. A recurrent layer adds an output o_l to the residual h_l, computing it from h_l and a *memory* m_l of the preceding tokens. Every intervention here replaces one of these at the final token: h_l in this section, o_l and attention-head outputs in Section 4, and m_l in Section 5.

Separating the key from its row. Across pairs the donor's key and the row of its completing fact vary independently, and garden, kitchen, and hall each take their turn in the key, position, and reference roles, so a place the model simply prefers cannot pass for key-based retrieval (Appendix Table 8). A generated story, with its own people, places and fact order, together with the donor and recipient prompts built from it, is one *identity*; the experiments score 200 identities each except where Appendix Table 4 says otherwise. An intervention can raise or lower every candidate at once, so subtracting the reference candidate's change removes what is common to all of them. Writing B_i, B_j, B_k and A for the key, position, reference and donor-answer candidates, garden, kitchen, hall and office in the example, the key effect κ, the position effect π and the donor-answer effect α are

$$\kappa = \Delta\ell(B_i) - \Delta\ell(B_k), \qquad \pi = \Delta\ell(B_j) - \Delta\ell(B_k), \qquad \alpha = \Delta\ell(A) - \Delta\ell(B_k). \tag{2}$$

Tasks and models. The generators produce 17-fact, two-hop problems in three forms. The *arrow* task uses arbitrary keys and numbers and the *binding* task names and apartment numbers; both carry the depth profiles and the targeted interventions. The *word* task uses objects, owners and places, and carries the memory experiments of Section 5. The depth panel covers twelve models: sequential hybrids, which alternate attention and recurrent layers; parallel hybrids, which combine them within a layer; and dense transformers. Targeted experiments use Nemotron-H, Granite-H, RecurrentGemma-9B and Nano. Every model runs as a raw completion model and must answer 60% of donor and recipient prompts before it is assayed. Section 5 runs its memory swap on seven hybrids, three of which do not read and count as part of its result; Granite-H is not among them, since it does not reach that gate in the memory roles (Appendix F.3). Appendix A lists the models, the identity pools and their roles.

Relation to prior work. Feng & Steinhardt (2024) identify binding representations, and Gur-Arieh et al. (2026) separate positional, lexical and reflexive retrieval with counterfactual pairs. We add a reference candidate and independent variation of key and position, to follow a *computed* key. For two-hop queries answered from parametric memory, Yang et al. (2024) and Biran et al. (2024) locate where the bridge entity is resolved and where the answer forms; our profiles measure both for a key computed from the context, one transplant giving them at every layer. Attention-query transfer reveals reusable filtering operations (Sharma et al., 2026). Work on hybrids removes a component, zeroing an output or dropping one of the two caches (Michalak & Abreu, 2025; Afendulev et al., 2026); Zani et al. (2025) find that a small subset of a hybrid's attention heads suffices for retrieval, which Section 4.2 reaches from the other side, by measuring what a transplanted key does. Removing a component leaves the model in a state it does not otherwise produce (Makelov et al., 2024; Grant et al., 2026); we substitute instead, always an activation the same model produced on a matched prompt, and Appendix I tracks how far the intervened trajectories fall from natural ones.

3 From key availability to answer resolution

Using the pairs and scores of Section 2, one transplant run per layer input measures what the remaining computation does with each state. The layer across which α rises most is the *resolving layer* (Section 3.2). In Nemotron-H's panel of Figure 2 the probe (green) first reads the key at L29, the key effect (orange) builds to the input of L40, and across L40 it collapses while the donor-answer effect (dotted) jumps.

3.1 An intermediate key becomes available before the answer

A linear ridge probe reads the key out of the same states well before the transplant converts it (Figure 2, green; Appendix C), and in all twelve models its largest rise falls across a layer that contains attention. Only the five sequential hybrids make that informative, since the others have attention in every layer. In those five attention sits in four to twelve of the layers, yet the rise lands on an attention layer in all five: picking a layer at random in each would do that about once in forty thousand, or once in three thousand counting the two Nemotron-H checkpoints, which place attention identically, as one. What happens between there and the layer that converts the key is the subject of Section 4.

In Nemotron-H and Granite-H the key effect rises to a *key-dominant* state at the input to the final attention layer (L40 and L35), κ positive and reliably larger than π, and the donor-answer effect takes over after it (Figure 2). At peak key availability Qwen2.5's state is *balanced*, both effects supported

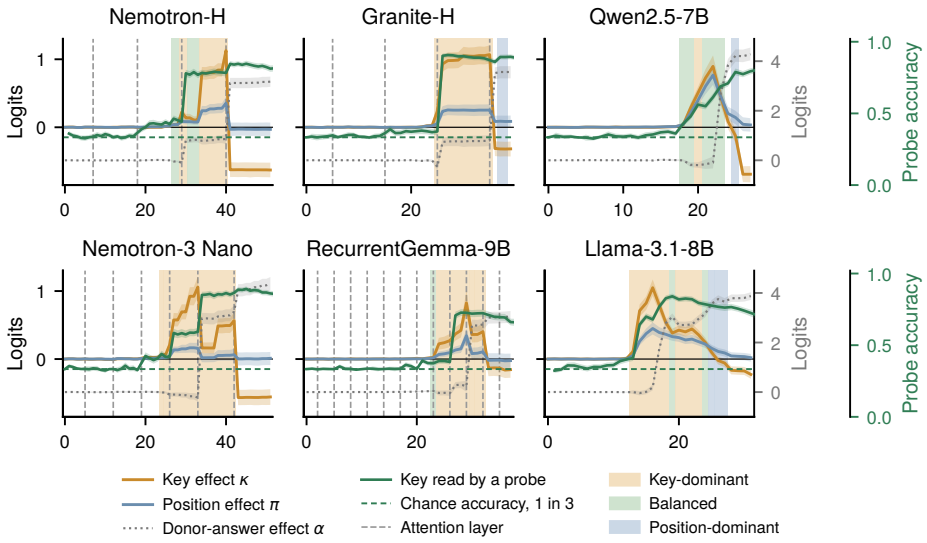


Figure 2: **Key availability rises before the main answer-conversion step.** Transplants at every layer input in four hybrids and two dense transformers; 200 identities with 97.5% identity-bootstrap intervals. The probe chooses among the donor identity's three keys, so its chance level is one in three. Dense panels carry no attention markers because every one of their layers has attention. The resolving layer's step makes the key effect negative in the three hybrids with four attention layers (Section 3.2; Appendix Table 16).

and their difference inside a prespecified band; its conversion spans several attention layers and Nano's has two stages (Section 3.2). Appendix C gives every layer's effects and class, Appendix H.1 the rules and the constants fixed in advance.

The transferred state carries a key, and a row pointer with it. Three tests find the key dominant and the row pointer supported (Table 1). In *rebinding*, moving Alice to a different place in the recipient while holding the donor state fixed moves the promoted answer with her. In *row mismatch*, separating the key from its row as in Section 2 leaves the effect following the key. In causal scrubbing (Chan et al., 2022), resampling donor states that share only the key keeps its answer ahead of the recipient's other candidates; sharing only the answer or only the row reverses that advantage. Appendix B gives each test's construction and its complete values.

	Nemotron-H	Granite-H	Nano
Top-1 among seven (%), chance 14			
rebinding	86	71	67
row mismatch	39	35	28
Causal scrubbing: donor shares only . . . (×)			
the key	3.42	3.18	3.29
its own answer	0.62	0.62	0.61
its row	0.47	0.51	0.52

Table 1: **The transferred state carries a key, and a row pointer with it.** The three tests of this section at the resolving layer's input (Section 3.2), 200 identities each. Top-1 is how often the key's place wins among seven candidates; the scrubbing rows give the factor by which the key's answer gains over the recipient's other candidates, $\times 1$ being no change. Constructions, intervals, and the preregistered form of the scrub test are in Appendix B.

3.2 Peak key availability marks the largest conversion step

Write α_l for the donor-answer effect of Equation 2 at layer input l, office against hall. Scoring against the reference candidate keeps conversion apart from mere suppression of the key candidate. The increment $\Delta_l = \alpha_{l+1} - \alpha_l$ compares separate transplants made immediately before and after layer l. The layer with the largest increment is the *resolving layer*; its input carries the peak key effect in every profile we measured (Appendix Table 15), the landmark the later sections use. It opens the band of layers carrying the rise in six models, falls inside it in three and closes it in three, so the alignment holds wherever in the band it sits (Appendix Table 16). In Nemotron-H the conversion begins at L29 and the key still builds to L40.

The step consumes the key. Across the resolving layer the key effect falls in all 23 profiles while the donor-answer effect rises. We call that decline consumption, in an operational sense: a linear probe still reads the key out of the state after the fall in all twelve models (Figure 2, green; Appendix C). How much falls follows what comes after the resolving layer, since the score asks what the remaining computation can still do with the key: the fall is complete in the three hybrids whose resolving layer is their last attention layer, and partial wherever attention follows, down to a quarter in Llama-3.1-8B (Appendix Table 16).

Key availability can rebuild between readers. Nano's penultimate (L33) and final (L42) attention layers both convert the key; we call them its *readers*. On both tasks the key effect falls across L33, the resolving layer, rebuilds mainly at recurrent layer L37, and falls again across L42, in a second conversion that is smaller and starts below the profile's peak (Figure 2, bottom left; Appendix Table 14). The landmark therefore marks the largest conversion, and a key that a reader has already consumed can be resolved again. Section 4.3 tests whether L37 carries the key L42 converts.

3.3 Attention carries the conversion

Is the rise in the donor-answer effect concentrated in one attention layer or spread over many? Let $\mathcal{T}$ be the attention layers (every layer in dense and parallel models) and take each one's share of the summed positive increments, a negative increment getting no weight:

$$p_l = \frac{\max(\Delta_l, 0)}{\sum_{j \in \mathcal{T}} \max(\Delta_j, 0)}, \qquad C_{\text{attn}} = \max_{l \in \mathcal{T}} p_l, \qquad e^H = \exp\left(-\sum_{l \in \mathcal{T}} p_l \log p_l \right). \tag{3}$$

Here C_{attn} is the largest share and e^H, the exponential of the shares' Shannon entropy, is the effective attention count, as perplexity is an effective vocabulary size (Hill, 1973; Jost, 2006): two equal contributors give two, one gives one.

Across twelve models and two tasks the rise occupies a similar late band of depth, and inside it the conversion uses the attention available. Every sequential hybrid has two attention layers in the band and spreads the rise over **1.7–2.0** of them; the models with attention in every layer spread it over **2.9–8.0**, and even RecurrentGemma, with twelve attention layers, confines the rise to the two in the band (Figure 3; Appendix Tables 14 and 16). Against a uniform allocation over each model's own attention layers the ordering reverses, since the dense models use a small part of the attention they have and the hybrids nearly all of theirs, so the effective attention count reports where the attention sits rather than how concentrated a model is (Appendix Table 34).

What availability does not explain is who does the converting. Attention carries 95% of Nemotron-H's rise while occupying four of the 28 token-mixing layers among its 52, and no sequential hybrid leaves recurrent layers more than an eighth of it (Appendix Table 14). How much recurrence takes depends on the task: on the word task, where these layers can read a binding from memory (Section 5), attention's share in Nemotron-H falls to 0.86, and it returns on the numeric twin, where the swap finds no read (Appendix Table 11). Most of the conversion stays in attention on both tasks. Intervening at the final token does not decide this between the two: a recurrent layer after the intervened site combines the injected residual with a memory built from the recipient's own unmodified prefix (Equation 5), so it has the same opportunity to resolve the key that attention has, and Section 5 shows it takes that opportunity where the task allows. These comparisons span checkpoints trained differently; Section 4 shows that attention heads carry the lookup within each hybrid.

4 How recurrence and attention implement the handoff

The profiles locate a usable key and the layer where it becomes an answer. This section asks which components carry the key to the reader and which read it.

4.1 Recurrent outputs carry the key to the reader

Two donors differ only in their key, Alice or Bob, keeping office as the answer and the completing fact at the same row. In the Alice run we swap one recurrent layer's final-token output for Bob's, continue to the resolving layer's input, and transplant that state into the recipient, which puts Alice in

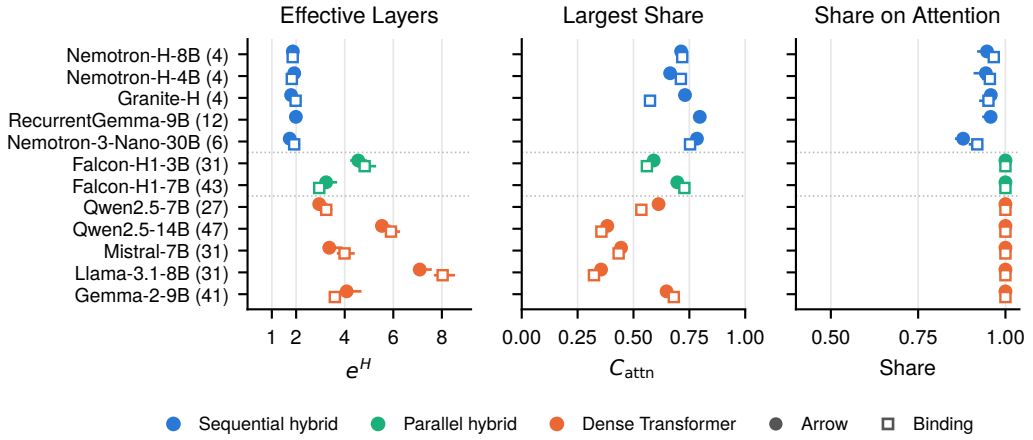


Figure 3: **The conversion uses the attention available in a late band of depth.** Each row is a model, filled circles the arrow task and hollow squares the binding task, with 95% identity-bootstrap intervals. Left: the effective attention count carrying the rise, e^H. Middle: the share of it carried by the single largest attention layer, C_{attn}. Right: the share falling on attention layers at all, which only the sequential hybrids make informative. Appendix Table 34 gives the first two against a uniform allocation.

the garden and Bob in the kitchen; if that output carries the key the lookup should move from garden to kitchen. Write κ_{Bob} and κ_{Alice} for the key effect of Section 2 read for Bob's place and for Alice's, after transplanting either donor's own state or the swapped one. We report both:

$$g = \frac{\kappa_{\text{Bob}}(\text{swap})}{\kappa_{\text{Bob}}(\text{Bob})}, \qquad h = \frac{\kappa_{\text{Alice}}(\text{swap})}{\kappa_{\text{Alice}}(\text{Alice})}, \tag{4}$$

Bob's lookup g, which Figure 4 plots, and Alice's lookup h (the guest and host lookups of Appendix D); an unchanged key gives $g = 0$ and $h = 1$. A separate depth profile ranks the recurrent layers before the reader by how much each raises the key effect. We swap the highest-ranked, the *nominated* layers, and as many nearby recurrent layers whose contribution there is near zero, the *matched* layers (Appendix D).

Swapping the nominated outputs switches part of the lookup to Bob in all four hybrids and reduces Alice's (Figure 4, left and middle). The two halves do not travel together. Matched layers, chosen because their contribution to the key is near zero, create some of Bob's lookup as well, and in Nemotron-H almost as much as the nominated layers do; what they do not do is remove Alice's, which they leave nearly intact where the nominated layers take it down (Appendix Table 17). So the nominated outputs are what the reader needs to find Alice, and part of what it needs to find Bob. As a control we swap between two donors that share the key but differ in the answer; the lookup then stays where it was, so the removal follows the key rather than the damage. How the key is carried differs by model: Granite-H relies on one nominated recurrent layer while its other two move neither lookup, and Nemotron-H and Nano spread it over several. The nominated layers straddle the attention layer where the probe first reads the key, so the swap shows that these outputs carry key-specific information into the reader without saying which of them prepared it.

4.2 A FEW ATTENTION HEADS READ THE KEY

At the resolving layers of Nemotron-H, Granite-H and Nano (Table 2) we intervene on attention-head outputs, on the data of Section 2, recording how each head's output changes under the transplant. *Injecting* a set's changes into an otherwise unmodified recipient asks whether those heads alone make garden rise; *clamping* all the layer's heads to their unmodified outputs asks what is left without them.

Eight heads, chosen by activation patching on separate examples (Appendix E), carry most of the transplant's effect and beat every random set of eight they were tested against; Granite-H spreads the reading over more heads. Clamping all heads removes the key effect wherever no later reader follows (Table 2). Giving the heads their donor-prompt outputs instead removes most of the key effect: they compute the answer from the recipient's facts rather than relay what they computed in the donor.

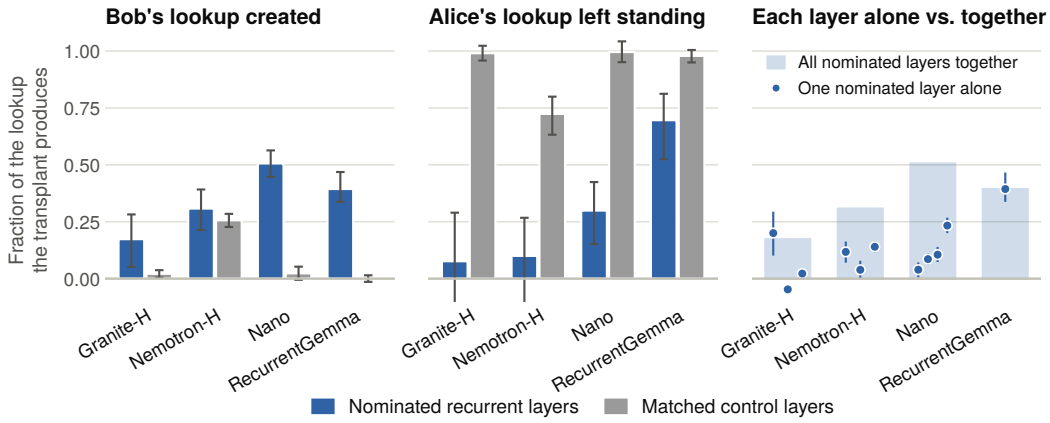


Figure 4: **Recurrent outputs carry the key to the reader.** Both lookups after the swap, each as a fraction of the lookup its own transplant produces, with 95% identity-bootstrap intervals; g and h of Equation 4. Left: Bob's lookup created, all nominated layers against all matched controls. Middle: Alice's lookup left standing, the same two sets, which is where they separate. Right: each nominated layer alone (points) against the joint swap (bar), which can fall below zero when one layer moves the lookup toward Alice. Appendix Table 17 reports both beside the redirection.

4.3 Recurrence can carry the key again between readers

Nano's key effect falls across its first reader (L33) and rebuilds before its final one (L42), most of it at the recurrent layer L37 (Figure 2). L37's contribution is hidden while the first reader still has the key, so we remove the key before L33: a second run asks about an object whose owner has no fact of their own, and its final-token outputs at the recurrent layers before L33 replace the host's. The prompt's facts stay as they are, so the layers after L33 can still find Alice, and a lookup that depends on the key survives at the final reader. Replacing L37's output as well removes 64% of it; the same replacement at a comparison layer in the same span removes none; and putting L37's undisturbed output back recovers 57% of what was lost. So L37 supplies part of what the final reader uses, and the reader's heads matter more once the early key is weakened (Appendix Table 19 and Figure 10).

	8-head injection	All heads clamped	Random sets beaten
Nemotron-H (L40)	≈100%	≈0%	100%
Granite-H (L35)	69–75%	≈0%	100%
Nano (L33)	93–96%	36–39%	100%

Table 2: **A few heads read the key.** Columns give the share of the transplant's key effect reproduced by injecting the eight heads' changes, the share surviving a clamp of all the layer's heads, and the fraction of random eight-head sets beaten; ranges span rebinding and row mismatch (Appendix Tables 20–22).

5 Recurrent memory can retrieve after the last attention layer

In the hybrids studied here, two to eight recurrent layers follow the last attention layer and can no longer call it (Appendix Table 29). Each keeps a memory m_l of the preceding tokens (Section 2), which in their Mamba-2 layers is a decayed sum of outer products. Dropping the layer index, the memory after token t and its reading are

$$m_t = a_t m_{t-1} + B_t x_t^\top, \qquad y_t = m_t^\top C_t = \sum_{s \le t} \Big(\prod_{r=s+1}^{t} a_r \Big) (C_t^\top B_s)\, x_s, \tag{5}$$

(Dao & Gu, 2024): each earlier token writes a value x_s under a key B_s, the decay factors a_r weight the earlier writes, and the final token retrieves the values whose keys match the query C_t it computes, an associative memory in the fast-weight sense (Schlag et al., 2021). Replacing m_l changes what is stored under which key and leaves that query untouched, so a hybrid in which these layers can look up a binding has a second retrieval route. We test it by exchanging two people's places in the word

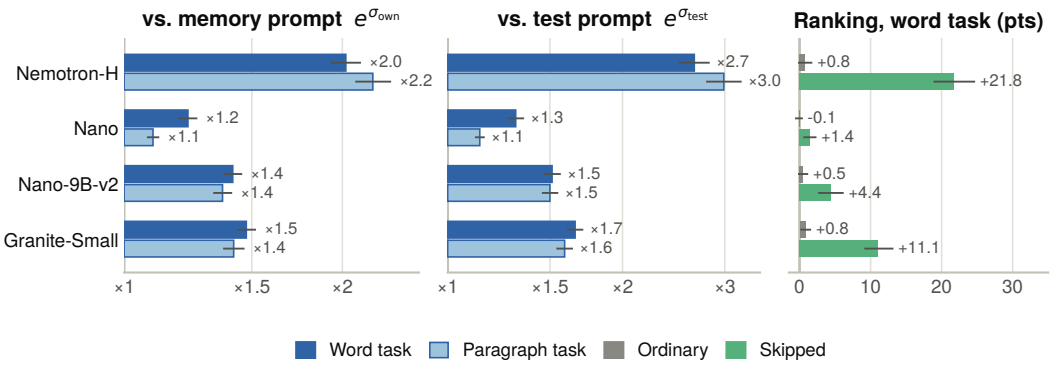


Figure 5: **Changing a stored location moves the answer in four hybrids.** Left and middle: geometric-mean factors for kitchen's probability against hall ($e^{\sigma_{\text{own}}}$) and against garden ($e^{\sigma_{\text{test}}}$), on logarithmic axes starting at $\times 1$. Both are read under ordinary execution; shade gives the task version, word or paragraph. Right: the change in how often the memory's place ranks first, on the word task, with color giving the incoming state, ordinary or skipped. Intervals are 95% over 200 identities; Appendix Table 29 lists every model and version tested.

task, which its facts allow without adding or removing a word. Earlier interventions on Nemotron-H's post-attention Mamba outputs motivate the test (Appendix F).

Writing a different fact into memory. The test prompt is the story of Section 2: the ball belongs to Alice, and Alice is in the garden. Let S be the recurrent layers after the last attention layer. We run the test prompt twice and, at its final token, install in every $l \in S$ the memories of a separate *memory prompt*, $m_l \leftarrow m_l(x^{\text{ctrl}})$ in one run and $m_l \leftarrow m_l(x^{\text{trans}})$ in the other. Everything else comes from the test prompt: its tokens, its question, and the residual entering S. Both memory prompts ask about Carol's cup and answer hall; they differ only in exchanging Alice's and Bob's places, so the transposed one puts Alice in the kitchen. Reading Alice's binding predicts kitchen and copying the memory prompt's own answer predicts hall, and we require kitchen to gain on both hall and garden, the test prompt's own answer. For each candidate place c, $\Delta\ell(c)$ is its transposed-minus-control logit change, which cancels what is common to both:

	Test	Control	Transposed
Alice	Garden	Garden	Kitchen
Bob	Kitchen	Kitchen	Garden
Carol	Hall	Hall	Hall
Question	Where is the ball?	Where is the cup?	Where is the cup?
Own answer	Garden	Hall	Hall

$$\sigma_{\text{own}} = \Delta\ell(\text{kitchen}) - \Delta\ell(\text{hall}), \qquad \sigma_{\text{test}} = \Delta\ell(\text{kitchen}) - \Delta\ell(\text{garden}). \tag{6}$$

Exponentiating a mean score gives the geometric-mean factor by which the exchange multiplies one candidate's probability relative to the other (Appendix F.3). Under *ordinary* execution the last attention layer has already put the test prompt's own answer several logits ahead, so a read of the memory can move the odds without deciding the answer. To ask what the read does when that lead is not already there, we run the swap a second time with the last attention layer's final-token update left out, the *skipped* state. The two differ only in the residual entering S at the final token.

The recurrent layers read the binding in four hybrids. The exchange promotes kitchen in all four models (Figure 5). In Nemotron-H it raises the kitchen-to-garden probability ratio by a geometric-mean factor of **2.7**, in every one of the 200 identities; the other three rise by smaller factors, each in at least 91% of identities taken one at a time. Ordinary execution changes the odds without changing the ranking. Skipping the last attention layer's final-token update removes the test answer's lead, and the read against that answer grows rather than weakening: the memory's place takes the top-1 answer from the test prompt's in **22** percentage points more identities in Nemotron-H, **11** in Granite-4.0-H-Small, **4.4** in Nemotron-Nano-9B-v2 and **1.4** in Nano (Figure 5, right), so the read does not run through the last attention layer. Each model repeats the result on 200 new identities with the facts written as one paragraph, and Appendix Table 29 adds a second sentence template and a version without the task's three distractor facts.

Controls. The control memory changes only the question the memory was built under, and installing it does not raise the memory prompt's own answer, hall, in three of the four models; it raises it slightly in the fourth, whose scores are net of that effect. The read is also specific to the binding the question asks about: in Nemotron-H, exchanging two places the question does not ask about moves the queried place's odds by $\times 1.06$, against $\times 2.7$ when the exchange reaches that binding, and Nano-9B-v2 and Granite-Small give the same picture. The answers also depend on the post-attention layers themselves (Appendix F.3 and F.4). Recurrent layers after the last attention layer can thus retrieve a binding from their own memory.

The read is addressed by the key, not by the row. With the test prompt's row order the two accounts predict the same place, since what the memory binds to the queried key also sits in that key's row. We separate them by rotating the key rows while keeping the bindings: the place bound to the queried key moves into another key's row, and the row that key occupies holds a third place. The answer follows the key in both models (Table 3), so the key addresses the content.

	Nemotron-H	Granite-Small
Key over row	$\times$ 1.9	$\times$ 1.4
Key over test prompt's	$\times$ 2.4	$\times$ 1.5
Row over test prompt's	$\times$ 1.3	$\times$ 1.1

Table 3: **The answer follows the key.** Geometric-mean factors after rotating the key rows, in ordinary execution; 200 identities per model. Intervals, the share of identities, and the construction are in Appendix Table 27.

Which models use it. Three kinds of null separate architecture, model and task. RecurrentGemma-9B keeps a vector state updated element by element, with no outer product for a query to address, so a key-addressed read has no direct counterpart there, though its output gate can still select per channel. Nemotron-H-47B and Nano-12B-v2 have Mamba-2 memories and show no read even so, while passing the same exact checks as the positive runs. Nano-9B-v2, pruned and distilled from Nano-12B-v2 (NVIDIA, 2025), reads where its parent does not: neither the pretraining corpus nor the architecture family settles which models use it. Nemotron-H also shows no read on the numeric version of the task, where the post-attention layers take a third as much of the answer conversion (Section 3.3), so reading and conversion vary together. Single-token room numbers leave the read where the multi-token version left it in Nemotron-H and Granite-Small (Appendix F.3), so length is not what the read needs of a value.

6 Discussion

A computed key moves through a sequential hybrid as a handoff. Earlier work showed that retrieval in hybrids depends on attention: zeroing attention-head outputs removes it, with no recurrent compensation (Michalak & Abreu, 2025), and pretrained hybrids delegate the aggregation step of in-context retrieval to their attention layers (Bick et al., 2025); in recurrent models associative recall is carried by components such as Mamba's short convolutions not by the recurrence itself (Arora et al., 2025). Our assay places these findings in one computation, locating where the key becomes usable and where it becomes the answer, and which components supply it and which read it. The depth profile agrees with the sites an activation-patching screen chose (Appendix E). Where the read appears, the recurrent state behaves like the fast weights of Schlag et al. (2021): written while the prompt is read, holding bindings under the keys that address them, and rewritable, since installing the memory of a prompt whose facts are exchanged moves the answer. What addresses it is the query the model computes, not a token in the prompt: the question names neither the person nor the place.

Scope. The tasks are generated fact lists read in raw completion mode, and each targeted intervention covers three to four hybrids; the assay needs only prompt pairs and a transplant, so natural text and other architectures are a direct extension rather than a tested one. The twelve checkpoints differ in training as well as in architecture, so the panel describes where the conversion happens and not what makes models differ. How far the reading concentrates, and whether the memory read reaches the model's first choice, are themselves model-dependent (Table 2; Figure 5). What a value must be for these layers to read it is open too: the read is strong on place words and near zero on room numbers, and neither single-token numbers nor number words recover it (Appendix F.3). Across these experiments a key becomes an answer through a handoff between recurrence and attention, and in four of the seven hybrids retrieval can continue past the last attention layer.

REPRODUCIBILITY STATEMENT

Every experiment runs from a configuration file that fixes the model and its revision, the intervention site, the identity pool and its seed, and the numerical thresholds, all written before the held-out sample was run; Appendix Table 4 lists each experiment with its data and its inferential role, and the supplement carries the configurations, their SHA-256 hashes, and the serialized prompts. The identity pools are generated from a seed and a tokenizer, so they can be rebuilt exactly. Appendix A gives the models, the accuracy gate every model passes before it is assayed, and the exact checks each run reports.

USE OF LARGE LANGUAGE MODELS

The authors used a large language model as a coding and writing assistant: it helped implement and run analysis scripts, draft and revise text, and check the paper against its own tables. Every experiment, result and claim was designed, verified and approved by the authors, who take full responsibility for the contents.

REFERENCES

Kirill Afendulev, Alexey Dontsov, Elena Tutubalina, and Anton Korznikov. What attention recalls and recurrence controls in hybrid language models, 2026. URL https://arxiv.org/abs/2609.04434.

Aryaman Arora, Neil Rathi, Nikil Roashan Selvam, Róbert Csordás, Dan Jurafsky, and Christopher Potts. Mechanistic evaluation of Transformers and state-space models. In *Mechanistic Interpretability Workshop at NeurIPS 2025*, 2025. URL https://openreview.net/forum?id=F62dVLJ8wk.

Aviv Bick, Eric P. Xing, and Albert Gu. Understanding the skill gap in recurrent models: The role of the gather-and-aggregate mechanism. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=hWYisuBbp7.

Eden Biran, Daniela Gottesman, Sohee Yang, Mor Geva, and Amir Globerson. Hopping too late: Exploring the limitations of large language models on multi-hop queries. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 14113–14130, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.781. URL https://aclanthology.org/2024.emnlp-main.781/.

Lawrence Chan, Adrià Garriga-Alonso, Nicholas Goldowsky-Dill, Ryan Greenblatt, Jenny Nitishinskaya, Ansh Radhakrishnan, Buck Shlegeris, and Nate Thomas. Causal scrubbing: A method for rigorously testing interpretability hypotheses. *AI Alignment Forum*, 2022. URL https://www.alignmentforum.org/posts/JvZhhzycHu2Yd57RN/causal-scrubbing-a-method-for-rigorously-testing. Non-peer-reviewed technical report.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality, 2024. URL https://arxiv.org/abs/2405.21060.

Danielle Ensign and Adrià Garriga-Alonso. Investigating the indirect object identification circuit in Mamba. In *ICML 2024 Workshop on Mechanistic Interpretability*, 2024. URL https://openreview.net/forum?id=lq7ZaYuwub.

Jiahai Feng and Jacob Steinhardt. How do language models bind entities in context? In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=zb3b6oKO77.

Atticus Geiger, Hanson Lu, Thomas F Icard, and Christopher Potts. Causal abstractions of neural networks. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=RmuXDtjDhG.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. Localizing model behavior with path patching, 2023. URL https://arxiv.org/abs/2304.05969.

Satchel Grant, Simon Jerome Han, Alexa R. Tartaglini, and Christopher Potts. Addressing divergent representations from causal interventions on neural networks. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=cZrTMqYVL6.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=tEYskw1VY2.

Yoav Gur-Arieh, Mor Geva, and Atticus Geiger. Mixing mechanisms: How language models retrieve bound entities in-context. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=UJ2UUjT2ko.

Mark O. Hill. Diversity and evenness: a unifying notation and its consequences. *Ecology*, 54(2): 427–432, 1973.

Lou Jost. Entropy and diversity. *Oikos*, 113(2):363–375, 2006.

Tom Lieberum, Matthew Rahtz, János Kramár, Neel Nanda, Geoffrey Irving, Rohin Shah, and Vladimir Mikulik. Does circuit analysis interpretability scale? evidence from multiple choice capabilities in Chinchilla, 2023. URL https://arxiv.org/abs/2307.09458.

Aleksandar Makelov, Georg Lange, Atticus Geiger, and Neel Nanda. Is this the subspace you are looking for? an interpretability illusion for subspace activation patching. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Ebt7JgMHv1.

Kurt Felix Michalak and Steven Abreu. Some attention is all you need for retrieval. In *Mechanistic Interpretability Workshop at NeurIPS 2025*, 2025. URL https://openreview.net/forum?id=SAYCGeL5aQ.

NVIDIA. NVIDIA Nemotron Nano 2: An accurate and efficient hybrid Mamba-Transformer reasoning model. Technical report, NVIDIA, 2025. arXiv:2508.14444.

NVIDIA, Aaron Blakeman, Aarti Basant, Abhinav Khattar, Adithya Renduchintala, Akhiad Bercovich, Aleksander Ficek, Alexis Bjorlin, Ali Taghibakhshi, Amala Sanjay Deshmukh, Ameya Sunil Mahabaleshwarkar, Andrew Tao, Anna Shors, Ashwath Aithal, Ashwin Poojary, Ayush Dattagupta, Balaram Buddharaju, Bobby Chen, Boris Ginsburg, Boxin Wang, Brandon Norick, Brian Butterfield, Bryan Catanzaro, Carlo del Mundo, Chengyu Dong, Christine Harvey, Christopher Parisien, Dan Su, Daniel Korzekwa, Danny Yin, Daria Gitman, David Mosallanezhad, Deepak Narayanan, Denys Fridman, Dima Rekesh, Ding Ma, Dmytro Pykhtar, Dong Ahn, Duncan Riach, Dusan Stosic, Eileen Long, Elad Segal, Ellie Evans, Eric Chung, Erick Galinkin, Evelina Bakhturina, Ewa Dobrowolska, Fei Jia, Fuxiao Liu, Gargi Prasad, Gerald Shen, Guilin Liu, Guo Chen, Haifeng Qian, Helen Ngo, Hongbin Liu, Hui Li, Igor Gitman, Ilia Karmanov, Ivan Moshkov, Izik Golan, Jan Kautz, Jane Polak Scowcroft, Jared Casper, Jarno Seppanen, Jason Lu, Jason Sewall, Jiaqi Zeng, Jiaxuan You, Jimmy Zhang, Jing Zhang, Jining Huang, Jinze Xue, Jocelyn Huang, Joey Conway, John Kamalu, Jon Barker, Jonathan Cohen, Joseph Jennings, Jupinder Parmar, Karan Sapra, Kari Briski, Kateryna Chumachenko, Katherine Luna, Keshav Santhanam, Kezhi Kong, Kirthi Sivamani, Krzysztof Pawelec, Kumar Anik, Kunlun Li, Lawrence McAfee, Leon Derczynski, Lindsey Pavao, Luis Vega, Lukas Voegtle, Maciej Bala, Maer Rodrigues de Melo, Makesh Narsimhan Sreedhar, Marcin Chochowski, Markus Kliegl, Marta Stepniewska-Dziubinska, Matthieu Le, Matvei Novikov, Mehrzad Samadi, Michael Andersch, Michael Evans, Miguel Martinez, Mike Chrzanowski, Mike Ranzinger, Mikolaj Blaz, Misha Smelyanskiy, Mohamed Fawzy, Mohammad Shoeybi, Mostofa Patwary, Nayeon Lee, Nima Tajbakhsh, Ning Xu, Oleg Rybakov, Oleksii Kuchaiev, Olivier Delalleau, Osvald Nitski, Parth Chadha, Pasha Shamis, Paulius Micikevicius, Pavlo Molchanov, Peter Dykas, Philipp Fischer, Pierre-Yves Aquilanti, Piotr Bialecki, Prasoon Varshney, Pritam Gundecha, Przemek Tredak, Rabeeh Karimi, Rahul Kandu, Ran El-Yaniv, Raviraj Joshi, Roger Waleffe, Ruoxi Zhang, Sabrina Kavanaugh, Sahil Jain, Samuel Kriman, Sangkug Lym, Sanjeev Satheesh, Saurav Muralidharan, Sean Narenthiran, Selvaraj Anandaraj, Seonmyeong Bak, Sergey Kashirsky, Seungju Han, Shantanu Acharya, Shaona Ghosh, Sharath Turuvekere Sreenivas, Sharon Clay, Shelby Thomas, Shrimai Prabhumoye, Shubham Pachori, Shubham Toshniwal, Shyamala Prayaga, Siddhartha Jain, Sirshak Das, Slawek Kierat, Somshubra Majumdar, Song Han, Soumye Singhal, Sriharsha Niverty, Stefania Alborghetti, Suseella Panguluri, Swetha Bhendigeri, Syeda Nahida Akter, Szymon Migacz, Tal Shiri, Terry Kong, Timo Roman, Tomer Ronen, Trisha Saar, Tugrul Konuk, Tuomas Rintamaki, Tyler Poon, Ushnish De, Vahid Noroozi, Varun Singh, Vijay Korthikanti, Vitaly Kurin, Wasi Uddin Ahmad, Wei Du, Wei Ping, Wenliang Dai, Wonmin Byeon, Xiaowei Ren, Yao Xu, Yejin Choi, Yian Zhang, Ying Lin, Yoshi Suhara, Zhiding Yu, Zhiqi Li, Zhiyu Li, Zhongbo Zhu, Zhuolin Yang, and Zijia Chen. Nemotron-H: A family of accurate and efficient hybrid Mamba-Transformer models, 2025. URL https://arxiv.org/abs/2504.03624.

Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear Transformers are secretly fast weight programmers. In *International Conference on Machine Learning*, 2021.

Arnab Sen Sharma, David Atkinson, and David Bau. Locating and editing factual associations in Mamba. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=yoVRyrEgix.

Arnab Sen Sharma, Giordano Rogers, Natalie Shapira, and David Bau. LLMs process lists with general filter heads. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=iPFlJESrsh.

Roger Waleffe, Wonmin Byeon, Duncan Riach, Brandon Norick, Vijay Korthikanti, Tri Dao, Albert Gu, Ali Hatamizadeh, Sudhakar Singh, Deepak Narayanan, Garvit Kulshreshtha, Vartika Singh, Jared Casper, Jan Kautz, Mohammad Shoeybi, and Bryan Catanzaro. An empirical study of Mamba-based language models, 2024. URL https://arxiv.org/abs/2406.07887.

Shenran Wang, Timothy Tin-Long Tse, and Jian Zhu. Understanding in-context learning beyond transformers: An investigation of state space and hybrid architectures. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 794–820, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.39. URL https://aclanthology.org/2026.findings-acl.39/.

Sarah Wiegreffe, Oyvind Tafjord, Yonatan Belinkov, Hannaneh Hajishirzi, and Ashish Sabharwal. Answer, assemble, ace: Understanding how LMs answer multiple choice questions. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=6NNA0MxhCH.

Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=EytBpUGB1Z.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=NG7sS51zVF.

Sohee Yang, Elena Gribovskaya, Nora Kassner, Mor Geva, and Sebastian Riedel. Do large language models latently perform multi-hop reasoning? In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 10210–10229, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.550. URL https://aclanthology.org/2024.acl-long.550/.

Davide Zani, Kurt Felix Michalak, and Steven Abreu. Contextual sparsity as a tool for mechanistic understanding of retrieval in hybrid foundation models. In *Sparsity in LLMs (SLLM): Deep Dive into Mixture of Experts, Quantization, Hardware, and Inference*, 2025. URL https://openreview.net/forum?id=TGWzg86kYv.

A EXPERIMENTAL SETUP AND NOTATION

Terminology. The paper and its tables use one vocabulary; the released runs and configuration files keep the field names the assay was built with, where the key is called *content* and the reference candidate the *neither* candidate, as in `content_vs_neither` and the scrub S_C.

Stage	Intervention	Data	Role
Flow localization	Residual patch	20 paired	Exploratory
Head ranking	Head-output patch	30 discovery + 300 held-out	Confirmatory
Fixed-donor crossing	Whole-state transplant	200 graphs / 600 crosses	Confirmatory
Fixed-recipient crossing	Whole-state transplant	200 graphs / 600 crosses	Confirmatory
Common assay: factorial	Whole-state transplant	200 identities / 1,800 cells	Confirmatory
Common assay: path patching	Head-delta injection, clamps	Same 200 identities	Confirmatory
Common assay: scrubbing	Matched resampling	Same 200 identities	Confirmatory
Cross-model panel	Prespecified screen + held-out run	Six checkpoints	Confirmatory
Granite-H factorial	Whole-state transplant, pre-L35	Same 200 identities	Confirmatory
Granite-H route and scrubs	Head-delta injection, clamps, causal scrubbing	Same 200 identities; 30 discovery	Confirmatory
Depth profile	Whole-state transplant at every layer	Same 200 identities, three models	Descriptive
Second prompt format	Whole-state transplant, pre-L39/L40	200 identities, second-format pool	Confirmatory
Probe reference	Ridge probes at every layer	240/80/200 identities	Descriptive
Panel profiles	Whole-state transplant at every layer	Same 200 identities, seven further checkpoints	Confirmatory (thresholds)
Shifted transplant	Whole-state transplant one layer later	Same 200 identities, three models	Confirmatory
Binding task	Whole-state transplant at every layer	200 identities, binding pool, full panel	Confirmatory (class; thresholds)
Peak alignment	Compare key peak and largest answer-conversion increment	23 profiles across twelve models	Descriptive; repeated on Nano
Writer swap	Recurrent-output replacement before attention	200 identities, four hybrids	Confirmatory
Suffix disruption and rescue	Mamba-output clamp and cross-donor rescue	200 identities in each of two word pools	Confirmatory
Recurrent-memory read	Recurrent-state replacement after attention	200 fresh identities per model and wording	Confirmatory
Memory-swap trajectory audits	Recipient-reference tail rates, wording-matched	Same 200 identities per template; 800-identity references	Descriptive
Output dependence	Final-token Mamba-output replacement, no transplant	100 fresh identities; 100-identity calibration	Confirmatory

Table 4: Experiments, data, and inferential roles.

Input matrices and the running example. Each prompt renders 17 facts as an ordered list, so a row index is prompt order, not graph distance. Figure 1b renames one saved assay pair (cell `mvp17-test-0000-oa-c0-a1-p1`) with names and places for readability; the figure source records the renaming, and the matrix below shows the pair.

	Donor facts		Recipient facts	
1	UBV	32	RAY	35
2	QNV	JTJ	WNV	BDH
3	EUG	36	CFG	32
4	CEZ	WNV	NOG	YED
5	VFS	31	UBV	39
6	IMS	YED	CEZ	WNV
7	BDH	JCF	IMS	CFG
8	MJH	RAY	GWY	TUV
9	TUV	38	SBW	34
10	GWY	TUV	QNV	JTJ
11	RAY	35	EUG	36
12	SBW	34	VFS	31
13	KOD	EUG	BDH	JCF
14	CFG	39	TUV	38
15	XXS	OVJ	XXS	OVJ
16	WNV	BDH	KOD	EUG
17	NOG	CFG	MJH	RAY
Query: NOG; answer: 39			Query: MJH; answer: 35	

Each row is rendered as `Key <key> -> Value <value>`. Both prompts begin with the same three solved one-step demonstrations: MIH$\rightarrow$ 90, EAF$\rightarrow$ 70, and IHE$\rightarrow$ 40. The task question is `Starting from <query>, follow mappings until reaching a number. Answer:`. The candidate answers are $\{31, 32, 34, 35, 36, 38, 39\}$. Keys and candidate values are renamed solely for presentation; all bindings and row positions are preserved.

Formal crossing and selectivity definitions. Every run is scored on the same seven candidate answers: three donor-answer values (the donor variants of the factorial share them), the three values the recipient binds to the keys under test, and the recipient's own answer; top-1 accuracy is taken among these seven. A candidate-level account H maps a donor–recipient pair to its predicted answer. A fully discriminating crossing satisfies

$$H_i(x_d, x_r) \neq H_j(x_d, x_r) \quad \text{for all } i \neq j. \tag{7}$$

For a predicted value B_j among three recipient-bound values,

$$S(B_j) = \Delta\ell(B_j) - \tfrac{1}{2} \sum_{k \neq j} \Delta\ell(B_k). \tag{8}$$

We call this mapping selectivity S^{map} in the fixed-donor crossing and key selectivity S^{con} in the fixed-recipient crossing. Tracking chooses the largest logit increase among these three values (chance 33%); top-1 chooses the highest final logit among all seven.

Notation. Table 5 collects the symbols of the main text.

Symbol	Meaning
$x_d, x_r; h_l(x)$	Donor and recipient prompts; the answer-position state entering layer l ("pre-Ll")
h_l, o_l, m_l	At the final token: the residual state entering layer l; the output a recurrent layer adds to it ($h_{l+1} = h_l + o_l$); the recurrent layer's memory, accumulated over the preceding tokens. Transplants replace h_l, writer swaps o_l, memory swaps m_l
$\Delta\ell(c)$	Change in the logit of candidate c when the donor's state replaces the recipient's (Eq. 1)
A; B_1, B_2, B_3; A_r	The donor's answer; the values the recipient binds to the three keys; the recipient's own answer
$\mathcal{A}, \mathcal{P}, \mathcal{K}$	The fixed-answer, position, and key causal models
$S(B_j)$; S^{map}, S^{con}	Selectivity of a candidate over the other two bindings; mapping (fixed-donor) and key (fixed-recipient) selectivity (Eq. 8)
κ, π	Key-vs-reference and position-vs-reference contrasts (Eq. 2)
Tracking, top-1	Rate at which the key-predicted candidate rises most among $B_{1:3}$; rate at which it ranks first among all seven candidates
F; P_{T}, P_{all}, $P_{\overline{\text{T}}}$, P_{B}	Whole-state transplant; head-output injection for the top-eight, all, other, and bottom-eight heads (Eq. 9)
R_{by}, D_{attn}, C_{T}	Residual bypass, donor-clamped heads, and top-eight heads clamped under F
α_l, Δ_l, C	The donor-answer effect (Eq. 2), its per-layer increment (Eq. 3), and its all-layer concentration (Appendix H.1)
S_C, S_A, S_P	Scrubs that preserve the key, the answer, or the row (Eq. 10)

Table 5: Notation.

Models. The primary model is Nemotron-H-8B. The crossed-assay panel was fixed before any transplant, and the depth-profile panel adds further checkpoints (Table 6); models below the fixed accuracy threshold are not analyzed further.

Checkpoint	Revision	Class	Layers	Task	Site	Outcome
nvidia/Nemotron-H-8B-Base-8K	94ea861e	H	52	Pass	pre-L40	Primary model
ibm-granite/granite-4.0-h-micro-base	372ede0b	H	40	Pass	pre-L35	Both crossings pass
ibm-granite/granite-4.0-h-small-base	4b4faa80	H	40	Pass	—	Memory swap only
Qwen/Qwen2.5-7B	d1497293	D	28	Pass	pre-L22/26[‡]	Balanced / reference
mistralai/Mistral-7B-v0.3	caa1feb0	D	32	Pass	pre-L19	Profile panel
Zyphra/Zamba2-2.7B	31afeeac	H	54	Fail[§]	—	Below accuracy gate
allenai/OLMo-2-1124-7B	7df9a825	D	32	Fail	—	Below accuracy gate
tiiuae/Falcon-H1-7B-Base	c9a4cbb9	P	44	Pass	pre-L29	Profile panel
nvidia/Nemotron-H-4B-Base-8K	faba3b73	H	52	Pass	pre-L40	Profile only
tiiuae/Falcon-H1-3B-Base	c096902c	P	32	Pass	pre-L21	Profile only
ibm-ai-platform/Bamba-9B-v2	b42852dc	H	32	Fail[§]	—	Below accuracy gate
Zyphra/Zamba2-7B	0ed988f3	H	81	Fail[§]	—	Below accuracy gate
Qwen/Qwen2.5-14B	97e1e763	D	48	Pass	pre-L36	Profile only
meta-llama/Llama-3.1-8B	d04e592b	D	32	Pass	pre-L16	Profile only
google/gemma-2-9b	33c19302	D	42	Pass[§]	pre-L28	Profile only
google/recurrentgemma-9b	141ed2b6	H	38	Pass[§] (arrow)	pre-L29	Third family; binding gate 55/53%
nvidia/NVIDIA-Nemotron-3-Nano-30B-A3B-BF16	bf77c317	H	52	Pass	pre-L33	Profile and targeted experiments; post-trained release, run without its chat template
nvidia/NVIDIA-Nemotron-Nano-9B-v2-Base	dc0661c8	H	56	Pass	—	Memory swap only
nvidia/NVIDIA-Nemotron-Nano-12B-v2-Base	78dc93a7	H	62	Pass	—	Memory swap only
nvidia/Nemotron-H-47B-Base-8K	81a3fb4f	H	98	Pass	—	Memory swap only

Table 6: Model panel. H: sequential hybrid; P: parallel hybrid (attention in every layer); D: dense Transformer. "Task: Fail" means unmodified accuracy fell below the fixed threshold; [§]evaluated with the model's beginning-of-sequence token after failing without it (Bamba-9B-v2 and both Zamba2 checkpoints fail both ways). For the profile panel the site column gives the layer with the largest key effect, and Table 14 gives each model's concentration and effective attention count. Crossed-assay sites were chosen by the prespecified screen, and every prespecified site and threshold is in the frozen configuration files whose SHA-256 the supplement lists; [‡]Qwen2.5's two sites were taken from the screen's descriptive audit and fixed before its factorial sample, and the depth sweep covers every layer.

Panel. Each model reads the unchanged prompts with its own tokenizer. Models trained with a beginning-of-sequence token are run with it (marked in Table 6), since that token serves as an attention sink (Xiao et al., 2024).

Binding task. Keys are first names; key-valued facts read "Alice's friend is Bob.", number-valued facts "Bob lives in apartment 39.", and the query follows the friends to an apartment number. Values stay two-digit, so the candidate readout is unchanged.

Pools and construction. The *crossed assay* evaluates each crossing on graphs never used for selection. The *common assay* draws disjoint training, validation, and test pools (Table 4). Within each identity a strength-two orthogonal array crosses the donor's key c, answer a, and row $p = (c + a) \bmod 3$, so every pair of factors is crossed once; the cells whose key and row differ enter the factorial, and a second arm keeps one donor state fixed across three permutations of the recipient's values. All candidates share a tens token, so we feed the shared prefix and read the unit-token logits. Absolute selectivities differ between the two assays (Table 7), so comparisons stay within an assay.

Quantity (logits)	Crossed assay	Common assay	Normalized
Recipient margin, A_r vs. runner-up	1.45	1.71 ± 0.21	1
Mapping selectivity, F	2.37	1.48 ± 0.29	0.88 ± 0.21
Key selectivity, F	1.02	1.14 ± 0.10	0.66 ± 0.09
Mapping selectivity, pre-L39	—	1.12 ± 0.20	0.67 ± 0.15
Key selectivity, pre-L39	—	0.85 ± 0.08	0.50 ± 0.07
F minus pre-L39, mapping	—	0.36 ± 0.09	—
F minus pre-L39, key	—	0.28 ± 0.03	—

Table 7: Scale across assays. Normalized values divide selectivity by the recipient's own A_r-versus-runner-up margin at the identity level. The pre-L39 whole-residual transplant is the matched-layer control.

Donor's key	Donor's decisive row at position		
	1 (Alice's fact)	2 (Bob's fact)	3 (Carol's fact)
Alice	Garden / Garden	Garden / Kitchen	Garden / Hall
Bob	Kitchen / Garden	Kitchen / Kitchen	Kitchen / Hall
Carol	Hall / Garden	Hall / Kitchen	Hall / Hall

Table 8: **The factorial separates the key from the position.** The recipient binds Alice, Bob, and Carol to garden, kitchen, and hall, whose facts occupy rows 1, 2, and 3. Each cell lists the location key-based retrieval predicts / the location row-based retrieval predicts. On the diagonal (gray) the two coincide; on the six off-diagonal cells they differ, and the remaining third location is the reference candidate.

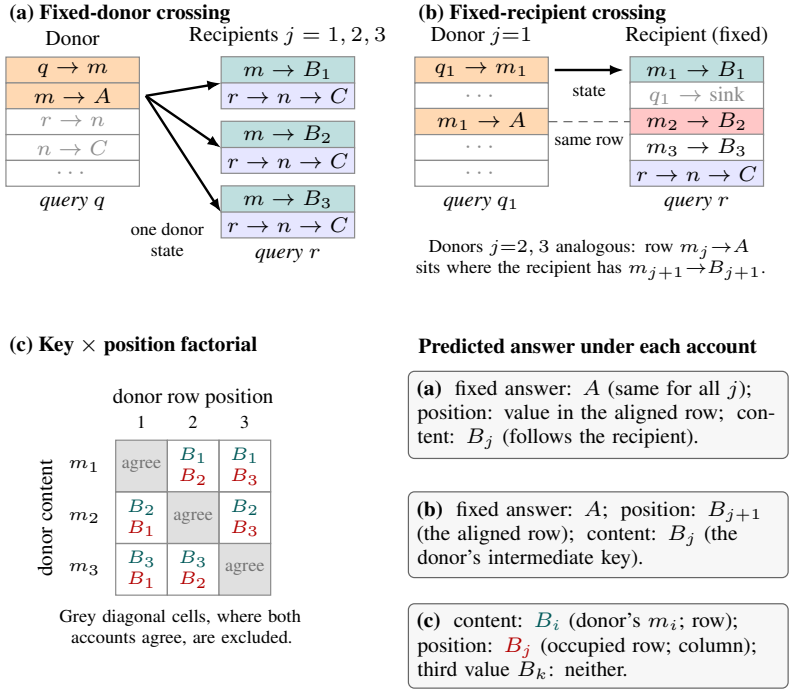


Figure 6: **Three crossed constructions.** (a) One donor state enters three recipients that bind the same key m to B_1, B_2, B_3. (b) Three donors whose queries reach different keys m_j enter one recipient; the donor row $m_j \to A$ is placed where the recipient holds $m_{j+1} \to B_{j+1}$, and the recipient maps the donor's query q_j away from B_j (to a dead end in most cells, Appendix A), so neither the query token nor the row position predicts B_j. (c) Donor key and donor row position vary independently; diagonal cells are excluded. Colors: orange donor path, teal recipient bindings under test, red adversarially aligned row, blue the recipient's own path; the lower-right panel lists each account's predicted answer.

B CROSSED ASSAY VALIDATION

Complete construction example. In the literal prompts behind Figure 1b, the donor's query reaches the key `CFG`, which the recipient still binds to a different value, while the donor's query token leads nowhere in the recipient, so the query token alone cannot predict the key's value. Following the donor's query through the recipient's own facts reaches the key's candidate in 57 of the pool's 1,800 cells, and excluding those leaves the key effect unchanged: 1.13 against 1.10 at Nemotron-H's pre-L40 site, and 1.07 against 1.07 at Granite-H's pre-L35. Serialized prompts for every construction are in the supplement.

		Nemotron-H, pre-L40	Granite-H, pre-L35
Fixed donor	Mapping selectivity (logits)	2.37 ± 0.21	2.48 ± 0.22
	Mapped vs. donor answer A (logits)	1.57 ± 0.19	1.67 ± 0.24
	Mapped vs. recipient answer A_r (logits)	4.17 ± 0.27	4.84 ± 0.31
	Tracking rate (%)	80.8 ± 4.5	86.0 ± 3.8
	Top-1 rate among seven (%)	85.8 ± 4.5	70.8 ± 5.5
Fixed recipient	Key selectivity (logits)	1.02 ± 0.12	0.99 ± 0.14
	Key vs. slot answer B_{j+1} (logits)	0.86 ± 0.14	0.89 ± 0.17
	Tracking rate (%)	52.8 ± 3.6	52.8 ± 3.5
	Top-1 rate among seven (%)	39.3 ± 3.9	35.2 ± 3.8
	Explicit-key reference, top-1 (%)*	98.8	99.3

Table 9: Crossed assay: 200 held-out graphs (600 crosses) per construction and model, with 95% intervals; chance tracking is 33.3%. The unmodified Nemotron-H recipient's margin between A_r and its runner-up is 1.45 logits. *Accuracy when the recipient is queried from m_j directly. Unmodified accuracies are in Appendix Table 10.

		Nemotron-H	Granite-H
Fixed-donor crossing	Unmodified donor accuracy (%)	88.5	82.0
	Unmodified recipient accuracy (%)	71.7	84.7
	Donor prefers A over A_r (%)	95.5	—
	Recipient prefers A_r over A (%)	92.2	—
Fixed-recipient crossing	Unmodified donor accuracy (%)	85.5	82.0
	Unmodified recipient accuracy (%)	82.0	84.7
	Donor prefers A over A_r (%)	93.7	—
	Recipient prefers A_r over A (%)	95.5	—
Granite-H L35 attention restored	Mapping selectivity, total / remaining	—	2.58 / -0.05
	Key selectivity, total / remaining	—	0.87 / -0.11
	Key vs. slot, total / remaining	—	0.70 / -0.12

Table 10: Unmodified accuracies (seven-candidate top-1 on the held-out graphs) and Granite-H layer-level mediation for the crossed assay. Granite-H used one held-out sample for both constructions; its mediation totals come from the mediation sample and differ slightly from Table 9.

C DEPTH PROFILES AND THE RESOLVING LAYER

Figure 2 shows the main depth profiles. The tables below give their contrasts and classifications at individual layer inputs, followed by the second-format and probe comparisons.

Nemotron-H	Attention	Recurrent, before L40	Recurrent, after L40
Arrow task	0.947	0.036	0.016 ± 0.005
Word task	0.862	0.046	0.092 ± 0.010
Numeric twin	0.936	0.036	0.028 ± 0.005

Table 11: Where the rise in the donor-answer effect falls, by task. Shares of the whole rise, with 95% identity-bootstrap intervals on the post-attention share, the one that separates the tasks: it is higher on the word task than on the arrow task by 0.075 ± 0.011 and than on the numeric twin by 0.064 ± 0.012. The word task is the one on which Section 5 finds the memory read and the numeric twin the one on which it does not. Within the word task's post-attention band the rise concentrates in L44 (0.044 of the whole, against 0.024, 0.017, 0.006 and 0.001 for L42, L46, L50 and L48), the layer whose output clamp costs the post-attention lookup most among all five clamped individually, on two identity pools (Appendix F).

Identities the model answers correctly. Every score in the paper averages over all 200 identities of its pool, whether or not the unmodified model answers them. Splitting Nemotron-H's pools by that leaves none of the effects smaller on the identities it answers. For the key effect at the pre-L40 input, where the donor and the recipient prompts must both be right and each is on 84% of identities, the 42 that qualify give 1.34 ± 0.22 and the other 158 give 1.07 ± 0.12, against 1.13 over all of them. For the memory read, where the test prompt and both memory prompts must be right and the three are right on 69% of prompts, the 35 that qualify give 1.08 ± 0.13 and 0.82 ± 0.11 and the other 165 give

0.96 ± 0.06 and 0.68 ± 0.05, against 0.98 and 0.70 over all. Nano, whose read is the smallest of the four, splits the same way in σ_{test} on the word task, where 98 of its 200 identities qualify (0.30 ± 0.05 against 0.24 ± 0.04 for the rest) and on the paragraph version, where 30 do (0.17 ± 0.04 against 0.12 ± 0.02). The effect is therefore present in both halves of both models, and the averages we report are the conservative ones; the subgroups are nested in the full pool, so these are descriptive splits rather than a test.

How much the alignment could have failed. The resolving layer and the peak of the key effect are read from the same profiles, so the alignment would be empty if the rise occupied a single layer. It does not: the span from a tenth to nine tenths of the cumulative rise covers three to twelve layers (median seven, Table 16), with two to nine attention layers inside it. Profiles of the same model on the two tasks share their rise spans, and three pairs of checkpoints are siblings, so we count the nine model families rather than the 23 profiles. A position-wise MLP or MoE layer maps the transplanted residual deterministically, so the transplants on either side of one are identical and its increment is exactly zero; such a layer cannot carry the largest step, and the null is taken over the layers that can. Drawing the largest step uniformly from the token-mixing layers in a family's span would place it at the key peak in 1.7 of the nine; drawing it from the attention layers in the span, the more generous null, gives 3.1. All nine coincide. What the alignment establishes is where the usable key is: in every profile we measured it is largest at the input of the layer that spends it, rather than at some earlier layer where the key is already available. Which components do the spending is what the interventions of Section 4 address.

Depth profile. We repeat F at the answer-position residual entering every layer of each model in Table 14, on the common-assay test identities, and classify each layer as in Section 2. The profile is descriptive: per-layer intervals carry no multiplicity adjustment, and no layer is selected by outcome.

Site	Key κ	Position π	$\kappa - \pi$	Class
Nemotron-H pre-L40	1.13 ± 0.12	0.36 ± 0.10	0.77 ± 0.11	Key-dominant
Nemotron-H pre-L39	0.84 ± 0.09	0.28 ± 0.07	0.56 ± 0.09	Key-dominant
Qwen2.5 pre-L22	0.86 ± 0.16	0.87 ± 0.14	-0.02 ± 0.14	Balanced
Qwen2.5 pre-L26	-0.63 ± 0.12	0.02 ± 0.11	-0.65 ± 0.09	Neither†
Granite-H pre-L35	1.07 ± 0.13	0.25 ± 0.09	0.82 ± 0.10	Key-dominant
Nemotron-H pre-L40 (colon format)	0.81 ± 0.12	0.31 ± 0.08	0.49 ± 0.09	Key-dominant

Table 12: Key $\times$ position factorial, in logits, on 200 held-out graphs (1,200 off-diagonal cells) per site. κ and π are the key-vs-reference and position-vs-reference contrasts of Eq. 2; they carry 97.5% intervals; the difference carries the 90% interval used for the prespecified ± 0.30 balanced region. Every key-dominant row is also dual-supported. The Granite-H class and the second-format class were predicted in advance. †The key is suppressed below the reference candidate and position does not reliably exceed it.

Model	No class	Balanced	Key-dominant
Nemotron-H	pre-L0–L26; pre-L41–L51 (fixed answer)	pre-L27, L28, L31–L33	pre-L29, L30, L34–L40
Granite-H	pre-L0–L24; pre-L36, L39 (fixed answer)	—	pre-L25§, L26–L35
Qwen2.5	pre-L0–L17; pre-L24, L26, L27 (fixed answer)	pre-L18, L19, L21–L23	pre-L20

Table 13: Prespecified class at every layer input. Position-dominant labels occur only in the fixed-answer regime (Granite-H pre-L37, L38; Qwen2.5 pre-L25), where the key lies below the reference candidate and the positional contrast is about 0.1 logits. §Key-dominant without dual support. Source and recipient seven-way accuracies were 84/84% (Nemotron-H), 77/81% (Granite-H), and 89/91% (Qwen2.5). Consecutive layers separated only by a position-wise MLP return identical values because the MLP is a deterministic function of the transplanted final-token state.

Second format. The factorial is regenerated under a second surface form (`XYZ: ABC` facts under a `Mappings:` header, a different question, and the prefix `Result:`); the prediction that pre-L40 keeps its class was fixed before the run and holds (Table 12).

The key is decodable long before it is read. Ridge probes fitted on the training identities and selected on the validation identities retrieve the intermediate key from the donor's residual well above chance ten layers before the causal effect peaks, and more often than the transplant tracks it (Figure 7); shuffled-label probes stay at chance, and decodability bounds a linear reader only. The key also survives the fall the resolving layer produces: at the layer input after it the probe recovers the key in 0.62 to 0.86 of identities across the twelve models, against a chance level of one in three.

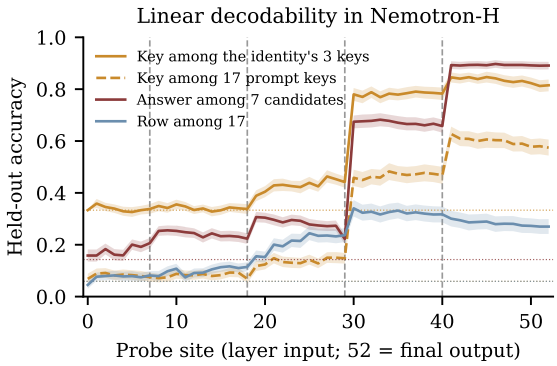


Figure 7: Ridge-probe decodability from the donor's answer-position residual at each layer input in Nemotron-H (site 52 is the final layer output), with 95% identity-clustered intervals; dotted lines mark chance and dashed lines the attention layers. Shuffled-label probes stay at chance at every site.

Model	Type	Layers	Key peak (class)	C, arrow	C, binding	e^H, arrow / binding	Attn. share
Nemotron-H-8B	Sequential hybrid	52 (4 attn.)	pre-L40 (key)	0.68 ± 0.03	0.69 ± 0.02	1.9 / 1.9	0.95
Nemotron-H-4B	Sequential hybrid	52 (4 attn.)	pre-L40 (key)	0.63 ± 0.03	0.68 ± 0.02	1.9 / 1.8	0.94
Granite-H	Sequential hybrid	40 (4 attn.)	pre-L35 (key)	0.70 ± 0.02	0.55 ± 0.02	1.8 / 2.0	0.96
Falcon-H1-3B	Parallel hybrid	32 (all)	pre-L21 (key)	0.59 ± 0.02	0.56 ± 0.02	4.6 / 4.8	1
Falcon-H1-7B	Parallel hybrid	44 (all)	pre-L29 (balanced)	0.70 ± 0.02	0.73 ± 0.01	3.2 / 3.0	1
Qwen2.5-7B	Dense	28	pre-L22 (balanced)	0.61 ± 0.02	0.54 ± 0.02	2.9 / 3.2	1
Qwen2.5-14B	Dense	48	pre-L36 (position)	0.38 ± 0.02	0.36 ± 0.01	5.5 / 5.9	1
Mistral-7B	Dense	32	pre-L19 (key)	0.44 ± 0.02	0.43 ± 0.02	3.4 / 4.0	1
Llama-3.1-8B	Dense	32	pre-L16 (key)	0.35 ± 0.01	0.32 ± 0.01	7.1 / 8.0	1
Gemma-2-9B	Dense	42	pre-L28 (key)	0.65 ± 0.02	0.68 ± 0.01	4.1 / 3.6	1
RecurrentGemma-9B	Sequential hybrid	38 (12 attn.)	pre-L29 (key)	0.76 ± 0.02	Gate fail	2.0 / —	0.96
Nemotron-3-Nano-30B	Sequential hybrid	52 (6 attn.)	pre-L33 (key)	0.69 ± 0.02	0.69 ± 0.02	1.7 / 1.9	0.88

Table 14: The model panel behind Figure 3: concentration C, the largest single layer's share of the rise in the donor-answer effect over all layers (95% intervals), on both tasks; the effective attention count e^H carrying that rise; attention's share of the rise (the summed positive increments at attention layers divided by the sum over all layers); and the site and class of the largest key effect κ on the arrow task.

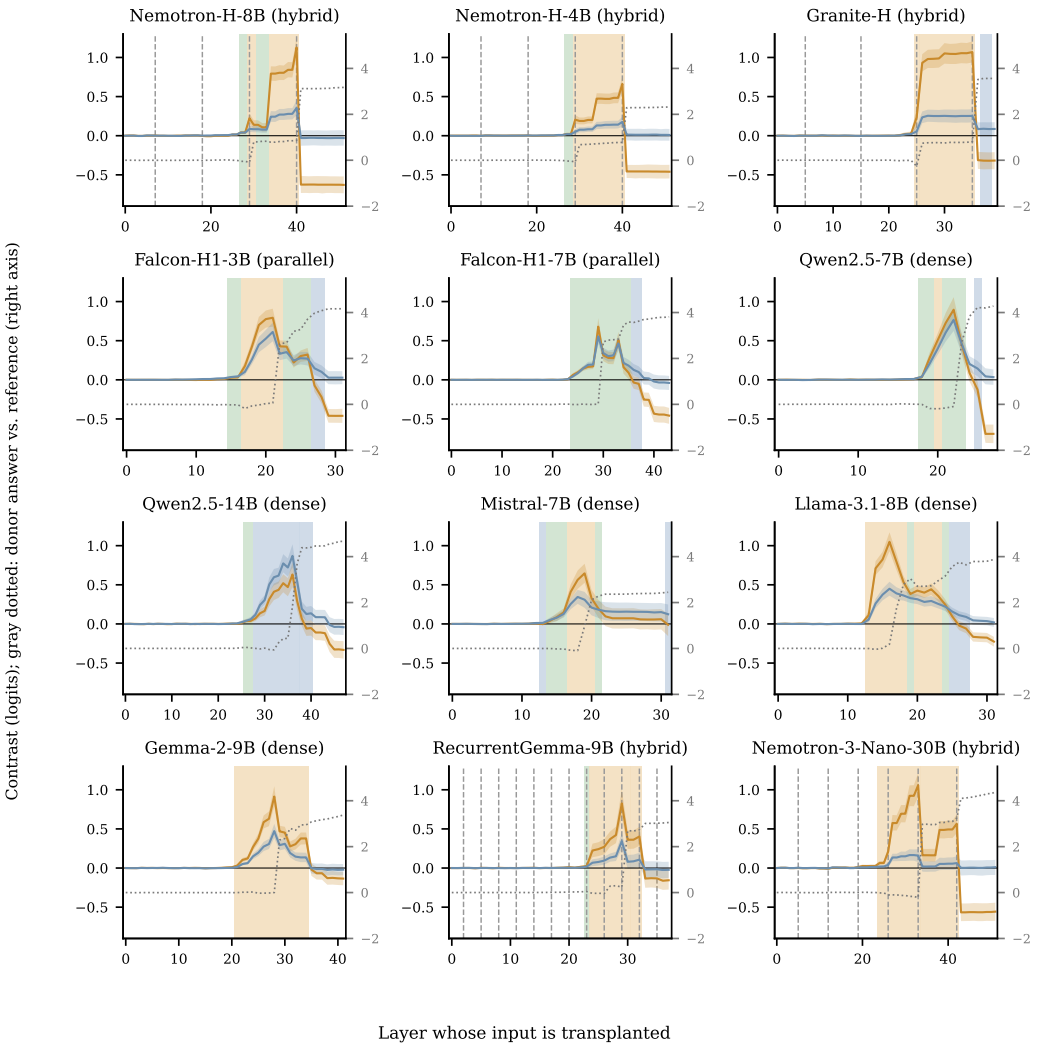


Figure 8: Depth profiles of the twelve models in Table 14, in the format of Figure 2. Dashed lines mark attention layers in the sequential hybrids; the parallel hybrids and dense models have attention in every layer.

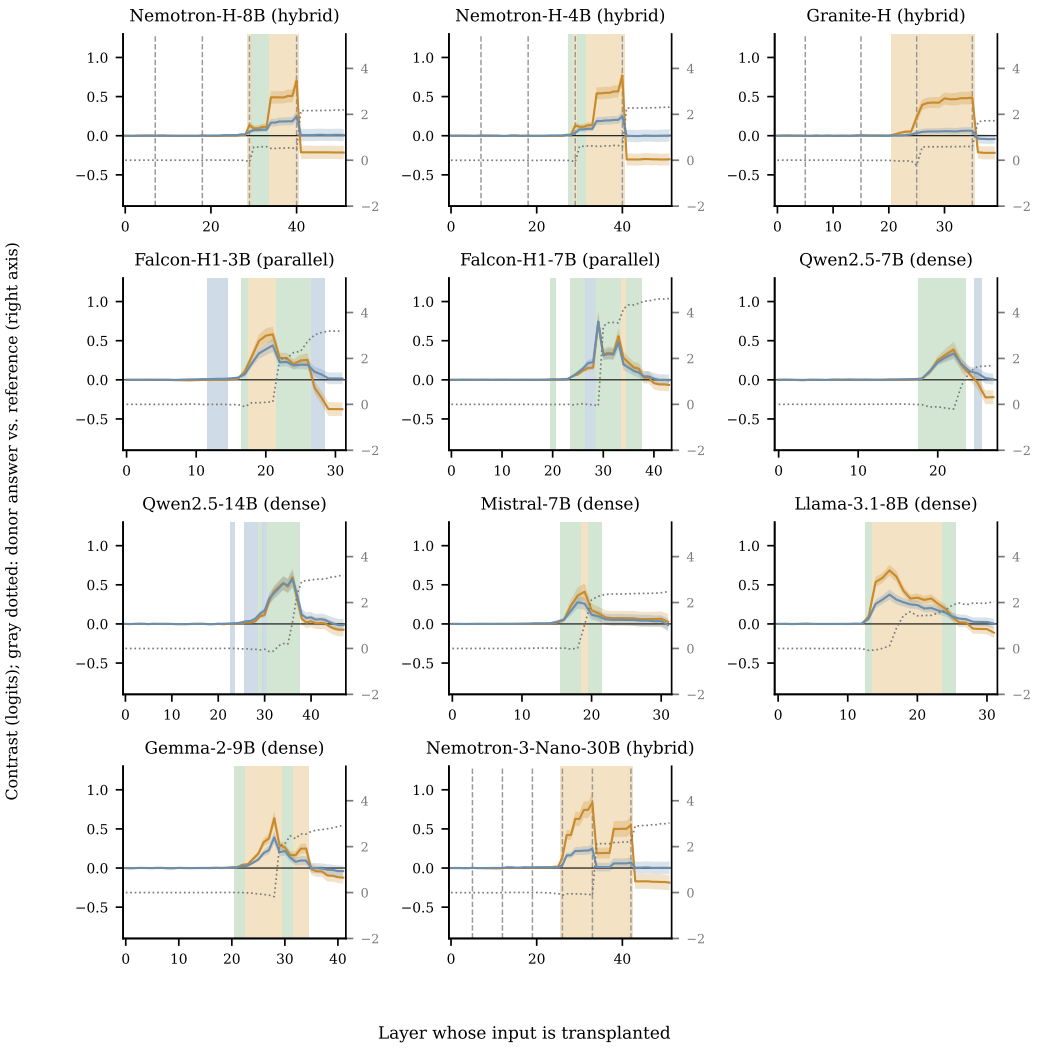


Figure 9: Depth profiles on the binding task of Section 2 for the models of Table 14 that passed the binding-task gate, in the format of Figure 8.

Peak alignment. In all 23 eligible profiles, the input to the layer with the largest donor-answer increment carries the peak key effect. The relationship was documented on the first 21 profiles before Nano was evaluated, and recurs on both Nano profiles. Resampling the identities of each profile (Table 15), the resolving layer never changes and the peak falls at its input in nearly every draw, under both contrasts. When two layer inputs carry the same key effect because only a position-wise MLP separates them, the earlier one is taken as the peak.

Task	Model	Peak = resolver	Coincide (%)	Peak margin over runner-up
Arrow	Nemotron-H-8B	L40	100.0	0.29 ± 0.04 (L38)
Arrow	Nemotron-H-4B	L40	100.0	0.18 ± 0.02 (L39)
Arrow	Granite-H	L35	99.4	0.01 ± 0.01 (L33)
Arrow	RecurrentGemma-9B	L29	100.0	0.40 ± 0.07 (L28)
Arrow	Nemotron-3-Nano-30B	L33	100.0	0.14 ± 0.04 (L31)
Arrow	Falcon-H1-3B	L21	97.8	0.02 ± 0.01 (L20)
Arrow	Falcon-H1-7B	L29	100.0	0.17 ± 0.05 (L33)
Arrow	Qwen2.5-7B	L22	100.0	0.17 ± 0.03 (L21)
Arrow	Qwen2.5-14B	L36	100.0	0.11 ± 0.06 (L34)
Arrow	Mistral-7B	L19	88.3	0.08 ± 0.07 (L18)
Arrow	Llama-3.1-8B	L16	100.0	0.23 ± 0.04 (L15)
Arrow	Gemma-2-9B	L28	100.0	0.28 ± 0.04 (L27)
Binding	Nemotron-H-8B	L40	100.0	0.19 ± 0.03 (L39)
Binding	Nemotron-H-4B	L40	100.0	0.20 ± 0.03 (L38)
Binding	Granite-H	L35	97.0	0.01 ± 0.01 (L33)
Binding	Nemotron-3-Nano-30B	L33	100.0	0.09 ± 0.02 (L32)
Binding	Falcon-H1-3B	L21	99.4	0.02 ± 0.01 (L20)
Binding	Falcon-H1-7B	L29	100.0	0.18 ± 0.06 (L33)
Binding	Qwen2.5-7B	L22	100.0	0.07 ± 0.02 (L21)
Binding	Qwen2.5-14B	L36	100.0	0.08 ± 0.04 (L34)
Binding	Mistral-7B	L19	97.9	0.05 ± 0.04 (L18)
Binding	Llama-3.1-8B	L16	100.0	0.08 ± 0.02 (L17)
Binding	Gemma-2-9B	L28	100.0	0.26 ± 0.04 (L27)

Table 15: Stability of peak alignment under resampling of identities (2,000 replicates per profile). "Coincide" is the share of replicates in which the layer input with the largest key effect is the input of the layer with the largest donor-answer increment. The margin is the peak key effect minus the largest key effect at any other layer input (the runner-up, in parentheses), with its 95% interval.

Attention layers in the conversion band. Table 16 gives, for each profile, the span of layers over which the donor-answer effect rises from 10% to 90% of its cumulative positive increment, the attention layers inside that span, and the effective attention count e^H of Eq. 3. The span occupies a similar band of depth in every model, and e^H tracks the attention layers inside it: the sequential hybrids have one or two there and use them, the dense models have more and spread the rise across them. RecurrentGemma-9B, with six attention layers in its second half, still confines the rise to two of them.

			Arrow task						Binding task					
Model	Attn. blocks	Late attn.	Rise span	Depth	Attn. in span	e^H	κ in	Consumed	Rise span	Depth	Attn. in span	e^H	κ in	Consumed
Nemotron-H-8B	4	2	L29–L40	0.57–0.78	2	1.9	1.13	1.55	L29–L40	0.57–0.78	2	1.9	0.70	1.30
Nemotron-H-4B	4	2	L29–L40	0.57–0.78	2	1.9	0.66	1.69	L29–L40	0.57–0.78	2	1.8	0.77	1.39
Granite-H	4	2	L25–L35	0.64–0.90	2	1.8	1.07	1.29	L25–L35	0.64–0.90	2	2.0	0.49	1.43
RecurrentGemma-9B	12	6	L26–L31	0.70–0.84	2	2.0	0.83	0.56	—	—	—	—	—	—
Nemotron-3-Nano-30B	6	3	L33–L42	0.65–0.82	2	1.7	1.06	0.85	L33–L42	0.65–0.82	2	1.9	0.83	0.77
Falcon-H1-3B	31	15	L21–L26	0.68–0.84	6	4.6	0.79	0.46	L21–L26	0.68–0.84	6	4.8	0.58	0.52
Falcon-H1-7B	43	21	L29–L33	0.67–0.77	5	3.2	0.68	0.54	L29–L33	0.67–0.77	5	3.0	0.74	0.55
Qwen2.5-7B	27	13	L22–L24	0.81–0.89	3	2.9	0.90	0.49	L22–L24	0.81–0.89	3	3.2	0.39	0.52
Qwen2.5-14B	47	23	L33–L37	0.70–0.79	5	5.5	0.63	0.54	L33–L37	0.70–0.79	5	5.9	0.60	0.47
Mistral-7B	31	15	L18–L20	0.58–0.65	3	3.4	0.65	0.49	L18–L20	0.58–0.65	3	4.0	0.41	0.57
Llama-3.1-8B	31	15	L16–L24	0.52–0.77	9	7.1	1.05	0.24	L16–L24	0.52–0.77	9	8.0	0.68	0.12
Gemma-2-9B	41	20	L28–L34	0.68–0.83	7	4.1	0.91	0.49	L28–L34	0.68–0.83	7	3.6	0.64	0.55

Table 16: Conversion band per profile: the layer span carrying the middle 80% of the cumulative positive rise in the donor-answer effect, its position as a fraction of depth, the attention layers inside it, and the effective attention count e^H; "κ in" is the key effect entering the resolving layer and "Consumed" its drop across that layer as a fraction of that value. "Late attn." counts attention layers in the second half of the network. RecurrentGemma-9B's binding run failed its gate.

Repeated conversion in Nano. On both tasks Nano's key effect peaks at the input of attention layer L33, which carries the largest share of the rise, with L42 carrying most of the rest; recurrent layer L37 rebuilds the key between them (Figure 2).

D RECURRENT WRITERS BEFORE ATTENTION

Writer swap. Table 17 reports the writer-swap experiment on 200 fresh identities disjoint from the common assay. Each identity has one host donor and guests with the host's row layout: key-swap guests matched on the answer and the decisive row but reaching a different intermediate key, and answer-swap guests matched on the key but with a different answer. The host is run with a guest's final-token output of a recurrent layer in place of its own, the run continues to the resolving layer's input, and that state is transplanted into the recipient; the same is done for matched low-increment layers, for the nominated writers together, and for the guest's whole residual after the writer. Because a redirection fraction cannot tell a removed host lookup from a created guest lookup, the table reports the guest and host lookups separately, the shared-key lookup retained under answer swaps, and the host's own logit shift. Heads-only and heads-clamped rows use the frozen head sets of Section 4.2 and Appendix E, except in Nano, whose head set at L33 was selected after this experiment ran, so its rows use every head of that layer. The residual row is an upper bound rather than a rival account: it carries everything the run has computed up to that layer, the nominated outputs among it, so it creates more of the guest's lookup than those outputs alone. What singles out the nominated recurrent layers from the other recurrent layers is the matched comparison, where they leave a tenth of the host's lookup and the matched layers leave seven tenths. In every model a recurrent-output swap creates part of the guest's lookup and same-key swaps retain the lookup. The frozen heads carry that effect in Nemotron-H, where the heads-only rows reproduce the guest lookup of the full swap, and part of it in Granite-H, where they carry 0.07 of the writer's 0.20. The heads-clamped rows are read from the guest and host columns rather than from redirection: clamping removes the lookup instead of moving it, leaving no guest lookup and almost no host lookup (0.00 and 0.08 in Nemotron-H), while redirection, a fraction between the host and guest transplants, rises because the run has left the host's end. In Granite-H the two head conditions do not divide the writer's effect between them: the clamped rows keep more guest lookup than the heads-only rows in all three (0.14 against 0.07 for the single writer, 0.13 against 0.05 for the writers together, and 0.14 against 0.08 for the residual), so part of what the writer creates reaches the answer without the frozen heads, and we do not have an account of that path.

Which recurrent outputs carry the key. Table 18 reports a prespecified follow-up on the same pairs: each nominated writer and each matched control swapped alone, each leave-one-out set, and the full sets, classified by two rules fixed before the run (*concentrated*: one writer carries at least half of the joint effect and removing it drops the set below half; *distributed*: every single writer below half and every leave-one-out set below the joint set). Granite-H meets the concentrated rule at L24. Nemotron-H meets the distributed rule, and its low-increment layers also create guest lookup, consistent with a key distributed over the recurrent layers.

Delaying the key. The no-key guest is the host prompt with its first-hop value replaced by a dead-end key. Its outputs at every key-writing layer before the resolving layer replace the host's, the run continues, and the resolver's and the next attention layer's inputs are transplanted into the recipient. The delay strongly attenuates the lookup at the resolver's input in Nano and RecurrentGemma, and in Nano leaves a lookup at the later layer's input that the later layer's heads carry (Figure 10a).

Interrupting and rescuing Nano's late supplier. The prespecified follow-up uses the delay experiment's Nano identities, early disruption, and reader sites. The profile nominates recurrent layer L37, with L39 as the comparison layer in the same span. We replace either layer's final-token output with its no-key donor output, or restore L37's undisturbed host output for rescue, and transplant the resulting pre-L42 state into the recipient. All four frozen questions are supported (Table 19): interruption reduces the later lookup by 0.154 ± 0.061 where the comparison layer costs nothing, rescue recovers 0.087 ± 0.050 of that and leaves the lookup within the third of the uninterrupted run that the rule fixed in advance allows, 0.067 ± 0.039 below the uninterrupted run, and clamping the later heads costs more after disruption than in the undisturbed host.

Condition	Redirection	Guest lookup	Host lookup	Shared key kept	Host frame
Nemotron-H-8B: reader L40, 8 of 32 heads, gate 84/84%					
Writer L33 swapped	0.04 ± 0.03	0.04 ± 0.04	0.95 ± 0.06	1.00 ± 0.02	-0.02 ± 0.04
Control L35 swapped	0.13 ± 0.02	0.14 ± 0.02	0.87 ± 0.05	1.00 ± 0.01	0.13 ± 0.04
All writers L28/L33/L39 swapped	0.58 ± 0.05	0.31 ± 0.09	0.10 ± 0.20	0.99 ± 0.03	0.32 ± 0.14
Control set L31/L35/L37 swapped	0.26 ± 0.03	0.26 ± 0.03	0.72 ± 0.09	1.00 ± 0.02	0.24 ± 0.05
Residual after L33 replaced	0.69 ± 0.02	0.74 ± 0.03	0.33 ± 0.17	0.99 ± 0.04	0.80 ± 0.13
Writer swap, heads only	0.04 ± 0.05	0.04 ± 0.03	0.95 ± 0.10	—	—
Writer swap, heads clamped	0.44 ± 0.07	0.00 ± 0.01	0.08 ± 0.10	—	—
All writers, heads only	0.55 ± 0.07	0.35 ± 0.10	0.22 ± 0.15	—	—
All writers, heads clamped	0.58 ± 0.07	0.11 ± 0.04	-0.09 ± 0.12	—	—
Residual, heads only	0.67 ± 0.04	0.70 ± 0.05	0.33 ± 0.14	—	—
Residual, heads clamped	0.51 ± 0.07	0.08 ± 0.04	0.01 ± 0.10	—	—
Granite-H: reader L35, 8 of 32 heads, gate 78/80%					
Writer L24 swapped	0.49 ± 0.07	0.20 ± 0.09	0.18 ± 0.23	1.01 ± 0.02	0.21 ± 0.13
Control L27 swapped	-0.01 ± 0.01	0.00 ± 0.01	1.02 ± 0.02	1.01 ± 0.01	0.00 ± 0.01
All writers L24/L26/L29 swapped	0.52 ± 0.07	0.17 ± 0.12	0.08 ± 0.27	1.01 ± 0.02	0.21 ± 0.16
Control set L27/L31/L33 swapped	0.02 ± 0.01	0.02 ± 0.02	0.99 ± 0.03	1.01 ± 0.02	0.04 ± 0.02
Residual after L24 replaced	0.57 ± 0.08	0.22 ± 0.11	0.02 ± 0.27	1.01 ± 0.02	0.19 ± 0.16
Writer swap, heads only	0.33 ± 0.08	0.07 ± 0.06	0.38 ± 0.13	—	—
Writer swap, heads clamped	0.61 ± 0.09	0.14 ± 0.07	-0.16 ± 0.20	—	—
All writers, heads only	0.35 ± 0.09	0.05 ± 0.07	0.30 ± 0.15	—	—
All writers, heads clamped	0.63 ± 0.10	0.13 ± 0.08	-0.21 ± 0.21	—	—
Residual, heads only	0.40 ± 0.08	0.08 ± 0.07	0.24 ± 0.14	—	—
Residual, heads clamped	0.63 ± 0.09	0.14 ± 0.07	-0.19 ± 0.20	—	—
RecurrentGemma-9B: reader L29, 16 of 16 heads, gate 80/75%					
Writer L28 swapped	0.34 ± 0.04	0.39 ± 0.06	0.70 ± 0.15	1.00 ± 0.02	0.29 ± 0.08
Control L22 swapped	0.01 ± 0.01	0.00 ± 0.01	0.98 ± 0.03	1.01 ± 0.02	0.00 ± 0.01
Residual after L28 replaced	Equals the guest transplant (the writer immediately precedes the reader)				
Writer swap, heads only	0.36 ± 0.07	0.36 ± 0.08	0.61 ± 0.14	—	—
Writer swap, heads clamped	0.45 ± 0.11	0.18 ± 0.04	0.27 ± 0.10	—	—
Nemotron-3-Nano-30B: reader L33, 32 of 32 heads, gate 82/78%					
Writer L30 swapped	0.08 ± 0.03	0.11 ± 0.04	0.94 ± 0.05	1.01 ± 0.02	0.09 ± 0.03
Control L21 swapped	-0.01 ± 0.01	0.00 ± 0.01	1.01 ± 0.03	1.00 ± 0.01	0.00 ± 0.01
All writers L25/L28/L30/L32 swapped	0.64 ± 0.04	0.51 ± 0.06	0.30 ± 0.14	0.99 ± 0.03	0.53 ± 0.11
Control set L14/L16/L18/L21 swapped	0.01 ± 0.03	0.02 ± 0.03	1.00 ± 0.05	1.00 ± 0.02	0.00 ± 0.03
Residual after L30 replaced	0.85 ± 0.02	0.88 ± 0.03	0.28 ± 0.14	1.00 ± 0.03	0.97 ± 0.16
Writer swap, heads only	0.18 ± 0.04	0.06 ± 0.03	0.73 ± 0.08	—	—
Writer swap, heads clamped	0.38 ± 0.06	0.05 ± 0.02	0.36 ± 0.06	—	—
All writers, heads only	0.65 ± 0.04	0.43 ± 0.07	0.22 ± 0.12	—	—
All writers, heads clamped	0.55 ± 0.07	0.18 ± 0.04	0.16 ± 0.07	—	—
Residual, heads only	0.78 ± 0.04	0.65 ± 0.05	0.19 ± 0.11	—	—
Residual, heads clamped	0.66 ± 0.07	0.37 ± 0.04	0.15 ± 0.07	—	—

Table 17: Writer swap. Redirection of the recipient's lookup from the host's key to the guest's, which normalizes the guest-minus-host contrast between the two transplants (0 = host transplant, 1 = guest transplant), the guest and host lookups g and h of Equation 4, the shared-key lookup retained under answer-swap guests, and the host's own logit shift toward the guest key's value; 95% identity-bootstrap intervals over 200 identities.

Swap	Guest lookup	Host lookup	Paired guest-lookup difference
Nemotron-H-8B: writers L28/L33/L39, controls L31/L35/L37, reader L40; pattern: distributed			
L28 alone	0.12 ± 0.04	0.78 ± 0.08	vs. L31: 0.16 ± 0.05
Control L31 alone	-0.04 ± 0.02	0.98 ± 0.04	
L33 alone	0.04 ± 0.04	0.95 ± 0.06	vs. L35: -0.10 ± 0.05
Control L35 alone	0.14 ± 0.02	0.87 ± 0.05	
L39 alone	0.14 ± 0.03	0.86 ± 0.04	vs. L37: -0.03 ± 0.03
Control L37 alone	0.17 ± 0.02	0.87 ± 0.05	
All but L28	0.19 ± 0.05	0.79 ± 0.09	Joint minus set: 0.12 ± 0.09
All but L33	0.24 ± 0.06	0.64 ± 0.10	Joint minus set: 0.07 ± 0.06
All but L39	0.17 ± 0.08	0.37 ± 0.14	Joint minus set: 0.14 ± 0.03
All writers	0.31 ± 0.09	0.10 ± 0.20	
All controls	0.26 ± 0.03	0.72 ± 0.09	
Granite-H: writers L24/L26/L29, controls L27/L31/L33, reader L35; pattern: concentrated			
L24 alone	0.20 ± 0.09	0.18 ± 0.24	vs. L27: 0.20 ± 0.09
Control L27 alone	0.00 ± 0.01	1.02 ± 0.02	
L26 alone	-0.05 ± 0.01	1.06 ± 0.05	vs. L31: -0.04 ± 0.02
Control L31 alone	-0.01 ± 0.01	1.00 ± 0.01	
L29 alone	0.02 ± 0.01	1.02 ± 0.04	vs. L33: -0.01 ± 0.02
Control L33 alone	0.03 ± 0.01	0.97 ± 0.03	
All but L24	-0.03 ± 0.02	1.08 ± 0.06	Joint minus set: 0.20 ± 0.11
All but L26	0.24 ± 0.11	0.09 ± 0.26	Joint minus set: -0.07 ± 0.02
All but L29	0.15 ± 0.11	0.15 ± 0.25	Joint minus set: 0.03 ± 0.03
All writers	0.17 ± 0.12	0.08 ± 0.27	
All controls	0.02 ± 0.02	0.99 ± 0.03	

Table 18: Single-writer and leave-one-out swaps on the writer-swap pairs. Guest and host lookups as fractions of the guest and host transplants' (95% identity-bootstrap intervals); the last column gives the paired guest-lookup difference used by the classification rules.

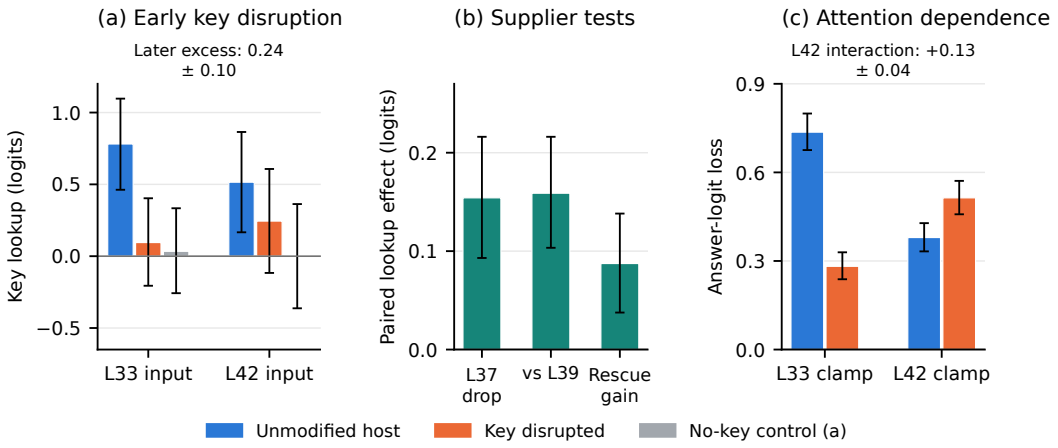


Figure 10: **A recurrent supplier contributes to Nano's later lookup.** (a) Host-key-vs-reference lookup at the two attention-layer inputs; the paired later excess over the no-key control is positive. (b) Paired supplier interruption, specificity, and rescue effects. (c) Answer-logit loss under head clamps before and after disruption; the annotated interaction tests increased L42 dependence. Intervals resample identities (95%).

Paired contrast	Estimate (95%)
Later excess over no-key	0.242 ± 0.097
Q1: L37 interruption drop	0.154 ± 0.061
L39 comparison drop	-0.005 ± 0.017
Q2: L37 versus L39 drop	0.159 ± 0.057
Q3: rescue improvement	0.087 ± 0.050
Rescue shortfall from uninterrupted	0.067 ± 0.039
L37/L41 joint interruption drop (secondary)	0.222 ± 0.072
L42 clamp loss, undisturbed host	0.379 ± 0.047
L42 clamp loss, disrupted host	0.514 ± 0.056
Q4: increase in L42 clamp loss	0.134 ± 0.037
L33 clamp loss, undisturbed host	0.736 ± 0.061
L33 clamp loss, disrupted host	0.282 ± 0.046
L37 interruption, share of the surviving lookup	64% [48, 85]
L39 comparison, share of the surviving lookup	-2% $[-12, 4]$
Rescue, share of the L37 interruption drop	57% [31, 79]

Table 19: Nano late-supplier follow-up on the original delay pool. Recipient key lookup is the host-key-vs-reference logit-change contrast; host clamp loss is the change in the host answer's logit. All entries are paired effects with 95% identity-bootstrap intervals. Q3 also has a separately frozen point-tolerance criterion, described in the text. The last three rows are the ratios the main text quotes, resampled as ratios, so their intervals are asymmetric.

E THE ATTENTION READER

Nominating the site and the heads. Sites and heads were nominated by activation patching on separate discovery examples: a clean activation is restored into a run whose target value is corrupted, and the top-ranked heads at a layer are tested on fresh examples against random head sets of the same size. The site and the top-eight head set were fixed before any crossed experiment.

Path patching details. Under F we cache the output of every head at the resolving layer, and starting from the unmodified recipient we inject only the head-output changes produced under F for a chosen set T,

$$o_h \leftarrow o_h + \mathbf{1}[h \in T]\left(o_h^F - o_h^R\right), \tag{9}$$

where o_h^F and o_h^R are head h's outputs under the transplant and in the unmodified recipient. Two clamps complete the design: a residual bypass R_{by} holds every head at its recipient output, and a donor-heads control D_{attn} holds them at the output they produced in the donor prompt; R_{by} plus the all-head injection reproduces F exactly. Each head set is compared with a bank of random sets of the same size.

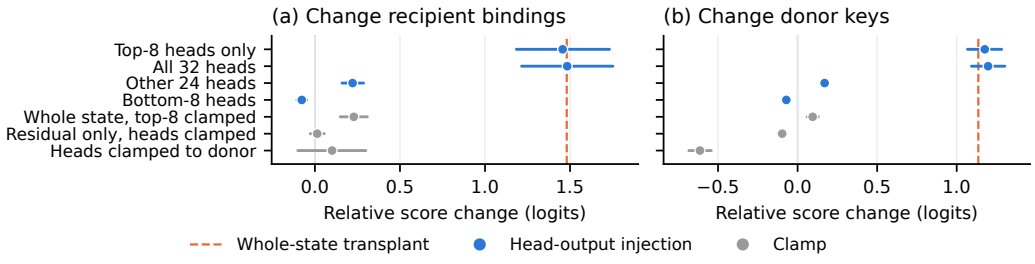


Figure 11: **Eight attention heads transmit most of the lookup effect.** Nemotron-H, L40, 200 identities, 95% intervals. The two panels change recipient bindings or donor keys, respectively; selectivity is the predicted value's score increase minus the mean increase of the other two recipient-bound values. Dashed lines show the whole-state transplant. Injecting the eight selected heads' changes nearly reproduces it; holding their outputs fixed sharply reduces it. Detailed values and Granite-H's comparison are in Appendix Tables 20 and 21.

Condition	What carries the transplant	Mapping selectivity	Key selectivity	Top-1 (%)
F	Whole state	1.48 ± 0.29	1.14 ± 0.10	42
P_{T}	Top-8 head outputs only	1.46 ± 0.28	1.18 ± 0.10	40
P_{all}	All 32 head outputs	1.48 ± 0.27	1.20 ± 0.10	41
$P_{\bar{\mathrm{T}}}$	Other 24 head outputs	0.22 ± 0.06	0.17 ± 0.03	4
P_{B}	Bottom-8 head outputs	-0.08 ± 0.03	-0.07 ± 0.01	3
R_{by}	Residual; heads clamped to recipient	0.01 ± 0.04	-0.10 ± 0.02	3
D_{attn}	Residual; heads clamped to donor	0.10 ± 0.20	-0.61 ± 0.07	2
C_{T}	Whole state, top-8 clamped	0.23 ± 0.08	0.09 ± 0.04	3

Table 20: Path patching at L40 in Nemotron-H on the common assay (200 identities), in logits with 95% intervals. Top-1 is the seven-way rate for the predicted candidate; the unmodified recipient scores 4%. $R_{\mathrm{by}} + P_{\mathrm{all}}$ reproduces F to zero error.

Condition	What carries the transplant	Mapping selectivity	Key selectivity	Top-1 (%)
F	Whole state	1.44 ± 0.29	1.09 ± 0.10	38
P_{T}	Top-8 head outputs only	1.00 ± 0.20	0.81 ± 0.08	25
P_{all}	All 32 head outputs	1.42 ± 0.27	1.15 ± 0.10	41
$P_{\bar{\mathrm{T}}}$	Other 24 head outputs	0.52 ± 0.11	0.41 ± 0.05	8
P_{B}	Bottom-8 head outputs	-0.14 ± 0.04	-0.12 ± 0.02	4
R_{by}	Residual; heads clamped to recipient	0.04 ± 0.06	-0.07 ± 0.04	3
D_{attn}	Residual; heads clamped to donor	0.06 ± 0.18	-0.36 ± 0.08	3
C_{T}	Whole state, top-8 clamped	0.57 ± 0.14	0.35 ± 0.06	8

Table 21: Path patching at L35 in Granite-H on the common assay, in the format of Table 20. Head sets were ranked on a 30-identity discovery pool and frozen before any test identity; the random bank holds 255 eight-head sets. The top-eight route carries 69% $\pm$ 6 of F's mapping and 75% $\pm$ 4 of its key selectivity, the rest spread across the other heads; it beats the bottom eight by 1.14 and 0.93 logits and every random set ($p = 1/256$); $R_{\mathrm{by}} + P_{\mathrm{all}}$ reproduces F to zero error.

Condition	What carries the transplant	Mapping selectivity	Key selectivity	Top-1 (%)
F	Whole state	1.18 ± 0.25	1.11 ± 0.13	28
P_{T}	Top-8 head outputs only	1.14 ± 0.26	1.04 ± 0.13	26
P_{all}	All 32 head outputs	1.01 ± 0.23	0.93 ± 0.12	18
$P_{\bar{\mathrm{T}}}$	Other 24 head outputs	-0.14 ± 0.04	-0.12 ± 0.02	3
P_{B}	Bottom-8 head outputs	-0.17 ± 0.05	-0.14 ± 0.02	3
R_{by}	Residual; heads clamped to recipient	0.43 ± 0.10	0.43 ± 0.06	4
D_{attn}	Residual; heads clamped to donor	0.33 ± 0.13	0.16 ± 0.06	2
C_{T}	Whole state, top-8 clamped	0.26 ± 0.08	0.26 ± 0.04	4

Table 22: Path patching at L33 in Nemotron-3 Nano on the common assay, in the format of Table 20. Head sets were ranked on a 30-identity discovery pool and frozen before any test identity (L33 heads 7, 5, 11, 0, 4, 1, 15, 13). The top-eight route carries 96% $\pm$ 7 of F's mapping and 93% $\pm$ 4 of its key selectivity, exceeds the bottom eight by 1.31 and 1.18 logits, and exceeds every one of 63 random eight-head sets ($p = 1/64$). Because Nano has a second attention layer after L33 (L42), the residual bypass keeps part of the effect.

E.1 Intervention-family comparison

Causal scrubbing resamples each donor state from the two other cells that share one factor and differ in the other two, preserving the key (S_C), answer (S_A), or row (S_P). Preservation is the negative centered mean-squared error between the seven-candidate logit changes:

$$\mathrm{Pres}(S) = -\left\| \widetilde{\Delta\ell}_S - \widetilde{\Delta\ell}_F \right\|_2^2 / 7. \tag{10}$$

In all three hybrids, preserving the key preserves the transplant's pattern of score changes better than preserving the answer or the row (Table 23); for each cell, the two other cells that share exactly the preserved factor supply the resamples. Table 25 relates these interventions to earlier manipulations of in-context binding.

	Key's answer against the recipient's others (×)			Preservation advantage	
Model	shares the key	shares the answer	shares the row	over answer	over row
Nemotron-H	3.42 ± 0.35	0.62 ± 0.04	0.47 ± 0.02	0.72 ± 0.11	0.86 ± 0.11
Granite-H	3.18 ± 0.35	0.62 ± 0.04	0.51 ± 0.03	0.97 ± 0.12	1.08 ± 0.11
Nemotron-3 Nano	3.29 ± 0.42	0.61 ± 0.05	0.52 ± 0.04	1.14 ± 0.15	1.12 ± 0.14

Table 23: **Preserving the key better preserves the transplant's effect.** Left: after the donor state is resampled from donors that share the named factor, the factor by which the key's answer gains over the recipient's other candidates, the exponentiated contrast reported in Table 1, with 95% intervals; $\times 1$ is no change. Right: the preregistered test, the preservation advantages of key-preserving over answer- and row-preserving resampling (Eq. 10). Positive values indicate a reduction in centered mean-squared error relative to the original whole-state transplant.

Family	Unit	Learned d.o.f.	Held-out outcome on the common assay
Crossed transplant	Complete residual	0	Key-dominant, dual-supported; mapping 1.48, key 1.14
Path patching	Head-output route	0 (set prespecified)	Top-8 route carries 98–104% of F in Nemotron-H, 93–96% in Nano and 69–75% in Granite-H
Causal scrubbing	Factor-preserving resample	0	Key-preserving scrub preserves, answer- and row-preserving do not, in all three hybrids

Table 24: The three families on identical held-out identities. Method-specific questions differ, so the outcome column is not a ranking.

Study	Manipulation	What it identifies	What crossing adds
Feng & Steinhardt (2024)	Binding-ID vectors added to, or swapped between, entity–attribute pairs	A binding subspace shared across entities and its geometry	A reference baseline and a key × position factorial, so that "binding" splits into the bound value and the bound row, each with a preregistered bound
Gur-Arieh et al. (2026)	Counterfactual prompt pairs that vary one retrieval cue (positional, lexical, reflexive) with activation patching	Which retrieval mechanism a model uses at a site, and that the mechanisms mix	Computed-key assembly in hybrids compared with dense models; a reference baseline and prespecified factorial classification
Sharma et al. (2026)	Patching a predicate (filter) representation between list prompts	A general filter-head circuit that selects list items	The key and position of a bound variable rather than a predicate, and three intervention families compared on one assay

Table 25: Relation to prior manipulations of in-context binding. Earlier binding studies also use counterfactual pairs and whole-state interventions. Our comparisons extend that approach to computed keys, hybrid architectures, and shared outcomes across intervention families.

F Retrieval after the last attention layer

F.1 Shifted transplant

Shifted transplant. Moving the donor state one layer past the final attention layer removes the key effect on the arrow and binding tasks in Nemotron-H and Granite-H, while Qwen2.5, with attention after the shifted site, keeps it. A state inserted one layer later than it was captured is off-distribution by construction; the fraction of runs whose final-layer state falls in the tail of natural states stays close to that of unmodified recipients in the hybrids. On the word task part of the key effect survives past Nemotron-H's last attention layer, which motivates Section 5; its numeric twin, with room numbers in place of location words, keeps none (Figure 12a).

F.2 Recurrent outputs after the last attention layer

The post-attention lookup depends on Mamba outputs. Two prespecified follow-ups on fresh word-task identities test the shifted lookup of Nemotron-H (Figure 12). Holding the final-token outputs of the five post-attention Mamba layers at the unmodified recipient's values removes the lookup, and numeric twins show none to remove. The pair was nominated from the word-task depth profile on an earlier identity pool, where L42 and L44 carry most of the post-attention rise, and not from any single-layer result. Clamping the nominated pair L42/L44 removes more than a comparison

pair, and restoring those outputs from a different donor with the same key brings the lookup back, while a donor with a different key does not; all six frozen criteria pass, again on fresh graphs under a new template (Table 26). Every exact check is zero. Clamped one at a time, the five layers cost the shifted lookup 0.063, $\mathbf{0.224}$, 0.118, 0.018 and 0.084 logits on the first pool and 0.021, $\mathbf{0.094}$, 0.023, -0.000 and 0.003 on the fresh one, for L42, L44, L46, L48 and L50: L44 is the largest in both, and L48 costs nothing in either.

Condition or paired contrast	Original (95%)	Fresh template (95%)
Shifted lookup	0.418 ± 0.063	0.126 ± 0.066
All five outputs clamped	-0.012 ± 0.023	0.004 ± 0.032
L42/L44 clamped	0.111 ± 0.045	0.010 ± 0.049
L46/L48 clamped	0.290 ± 0.053	0.104 ± 0.057
Same-key, different-answer rescue	0.406 ± 0.057	0.117 ± 0.060
Different-key, same-answer control	-0.041 ± 0.050	-0.053 ± 0.046
Collective clamp drop	0.430 ± 0.052	0.122 ± 0.048
Nominated-pair drop	0.307 ± 0.037	0.116 ± 0.031
Pair specificity	0.179 ± 0.042	0.094 ± 0.030
Rescue improvement	0.295 ± 0.038	0.107 ± 0.029
Rescue key specificity	0.447 ± 0.063	0.170 ± 0.050

Table 26: Nemotron-H suffix localization and cross-donor rescue, and their prospective replication on fresh graphs under a new template. Key-vs-reference logit changes and paired differences, with 95% identity-bootstrap intervals (200 identities per pool). Clamps use recipient baselines; rescue/control use different shifted donors. All six primary criteria pass in each pool.

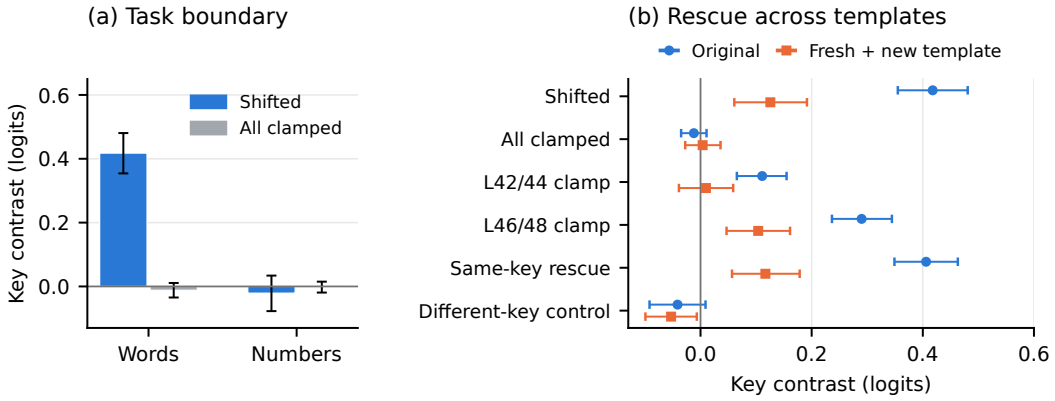


Figure 12: **Key-specific rescue of the post-attention lookup.** Nemotron-H, key effect κ in logits with 95% identity-bootstrap intervals. (a) Shifted lookup and collective Mamba clamp on fresh word/numeric twins. (b) The nominated pair, comparison pair, and cross-donor rescue in the original word pool and a fresh pool under a new template, with all sites frozen. Both support all six primary contrasts.

F.3 Recurrent-memory intervention

For each identity, three keys m_c are bound to three places B_c. For a fixed point $d \neq c$, a control memory prompt uses the recipient's facts and asks a two-hop question whose intermediate key is m_d and answer B_d; its transposed partner has the same question, answer, row layout, and values, but exchanges the places bound to the two other keys, so it binds the queried m_c to a third value, the *supplied* value. Every token except the final one is prefilled into the model's cache; at the final token only the recurrent states of the post-attention layers are replaced, while the residual entering them and the cached convolution histories belong to the run being tested. Swapping all post-attention memories moves the answer in both conditions, and so do smaller swaps (Table 28); the result replicates under a second template with every site, endpoint, and rule frozen. A trajectory audit covers every condition of the swap, the ordinary-regime ones included, and finds no excess final-layer atypicality in any of them: against the unmodified recipient each paired contrast is -0.01 ± 0.01, for the ordinary swaps as for the shifted. The one condition it cannot clear is the plain same-site transplant, whose interval still contains zero. In every run the single-token decode matches a full forward within the frozen bars, and every exact check is zero.

Granite-H. Granite-H carries two of the paper's targeted experiments but takes no memory role. It is not excluded by choice: on the two wordings screened it answers 53% and 58% of donor prompts, 38% and 46% of recipients, and 47% and 57% of memory prompts, so all three roles fall below the 60% gate every model passes before it is assayed (Section 2). A replication on this task setup would not be interpretable for it.

Key specificity. A preregistered control repeats the swap with the memory's fixed point set to the queried key m_c. Its transposed memory then binds m_c to the place the test prompt already binds it to and exchanges the places of the other two keys instead, so nothing the question asks about moves and a reader of the installed memory should leave the answer alone. In Nemotron-H on 200 identities the two exchanged places gain nothing over the queried answer under the shifted state: -0.005 ± 0.020 logits for the all-layer swap, and intervals containing zero for the pair and for L44 alone. In ordinary execution a small residual remains, 0.060 ± 0.021 for the all-layer swap and 0.042 ± 0.020 for L44, against 0.978 for the exchange that reaches the queried binding; it sits on one of the two exchanged places rather than both. Nano-9B-v2 and Granite-Small repeat the control on their all-layer swaps with the same shape: the exchanged places gain nothing under the shifted state, -0.004 ± 0.016 and 0.007 ± 0.020, and the same small residual remains under ordinary execution, 0.038 ± 0.013 and 0.034 ± 0.016.

Rotated key rows. The memories above keep the test prompt's row order, so the place a memory binds to the queried key also sits in that key's row and the two accounts predict the same answer. The crossed memory separates them: it holds the same exchanged bindings and writes them into key rows rotated by one, each row taking the bindings of the row below it and the last taking the first's, so the place bound to the queried key now sits in another key's row, and the row the queried key occupies holds a third place. Everything else follows the swap above. The control is the same unexchanged memory in the recipient's own row order, so the crossed memory differs from it in two ways at once, the exchange and the layout; what separates the two accounts is the comparison between the two places the crossed memory puts in play, both read against that one control. The answer follows the key, under both incoming states and in almost every identity (Table 27). The rotation costs the read little: against the test prompt's own answer the supplied place gains 1.10 logits when the memory keeps the recipient's row order and 0.89 when its rows are rotated. The run is the swap protocol of Section 5 on a fresh pool of 200 identities, with its sites, sample, gates and exact checks unchanged; the crossed memory and the key-versus-row contrast were fixed in its configuration before the run. That configuration fixed the shifted state as the second incoming state, where Section 5 reports the skipped one, so the two meet in ordinary execution, which is the column Table 3 quotes.

Model	Incoming state	Key over row	Key over test prompt's	Row over test prompt's	Identities, key > row
Nemotron-H	Ordinary execution	0.65 ± 0.04	0.89 ± 0.05	0.24 ± 0.03	98%
	Shifted	0.82 ± 0.05	1.00 ± 0.07	0.18 ± 0.03	100%
Granite-Small	Ordinary execution	0.31 ± 0.03	0.40 ± 0.02	0.09 ± 0.01	98%
	Shifted	0.38 ± 0.03	0.48 ± 0.03	0.10 ± 0.01	100%

Table 27: The crossed memory, which writes the same bindings into key rows rotated by one. Contrasts are logit changes of the crossed memory against the control memory, differenced between two candidates, with 95% identity-bootstrap intervals over 200 identities; the last column is the share of identities in which the place the key is bound to gains more than the place now in the key's row.

Template	Execution	All five memories	L44 memory only
Original	Shifted	0.331 ± 0.036 / 0.674 ± 0.065	0.187 ± 0.028 / 0.369 ± 0.042
Original	Ordinary	0.704 ± 0.045 / 0.978 ± 0.057	0.385 ± 0.034 / 0.537 ± 0.041
Fresh wording	**Shifted**	$\mathbf{0.111} \pm 0.022$ / $\mathbf{0.233} \pm 0.036$	$\mathbf{0.064} \pm 0.013$ / $\mathbf{0.118} \pm 0.020$
Fresh wording	**Ordinary**	$\mathbf{0.375} \pm 0.027$ / $\mathbf{0.530} \pm 0.035$	$\mathbf{0.202} \pm 0.018$ / $\mathbf{0.285} \pm 0.021$

Table 28: Recurrent-memory effects for all five layers and L44 alone. Each cell gives the supplied value over the memory prompt's own answer / over the recipient's value, in logits with 95% intervals. Bold marks the fresh-wording replication. The all-five-layer results also appear with separate contrast columns in Table 29.

Model (recurrent layers after last attention)	Task version	Accuracy (S / R / M, %)	σ_{own}: over the memory prompt's answer		σ_{test}: over the test prompt's answer		Consistent identities (%)	Read
			Shifted	Ordinary	Shifted	Ordinary		
Nemotron-H-8B (5 Mamba-2)	Word	71 / 68 / 69	0.33 ± 0.04	0.70 ± 0.04	0.67 ± 0.07	0.98 ± 0.06	100	Both conditions
	Story	64 / 61 / 63	0.11 ± 0.02	0.37 ± 0.03	0.23 ± 0.04	0.53 ± 0.03	99	Both conditions
	Paragraph	71 / 66 / 70	0.44 ± 0.04	0.79 ± 0.05	0.89 ± 0.09	1.09 ± 0.06	100	Both conditions
	Short word	76 / 74 / 78	0.33 ± 0.04	0.78 ± 0.04	0.68 ± 0.08	1.05 ± 0.06	99	Both conditions
	Numeric twin	88 / 87 / 86	0.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.01	0.00 ± 0.01	18	Not detected
	Numeric, one token	91 / 88 / 87	$0.01 \pm 0.01^{\dagger}$	0.01 ± 0.01	$0.02 \pm 0.01^{\dagger}$	0.01 ± 0.01	33	Ordinary
	Number words	85 / 82 / 84	$0.02 \pm 0.00^{\dagger}$	0.03 ± 0.00	$0.05 \pm 0.01^{\dagger}$	0.04 ± 0.01	60	Ordinary
Nemotron-3 Nano 30B (4 Mamba-2)	Word	86 / 84 / 87	0.13 ± 0.02	0.20 ± 0.03	0.27 ± 0.03	0.27 ± 0.03	91	Both conditions
	Story	75 / 69 / 72	$0.08 \pm 0.02^{\dagger}$	0.15 ± 0.02	$0.13 \pm 0.02^{\dagger}$	0.21 ± 0.03	86	Ordinary
	Paragraph	69 / 62 / 68	0.07 ± 0.02	0.09 ± 0.01	0.15 ± 0.01	0.13 ± 0.02	76	Both conditions
	Short word	86 / 85 / 89	0.13 ± 0.02	0.21 ± 0.02	0.23 ± 0.03	0.27 ± 0.03	88	Both conditions
Nemotron-Nano-9B-v2 (8 Mamba-2)	Word	88 / 86 / 86	0.18 ± 0.02	0.34 ± 0.02	0.37 ± 0.04	0.41 ± 0.03	99	Both conditions
	Paragraph	81 / 76 / 81	0.17 ± 0.02	0.31 ± 0.03	0.37 ± 0.03	0.41 ± 0.02	97	Both conditions
	Short word	90 / 88 / 90	0.15 ± 0.03	0.34 ± 0.03	0.29 ± 0.04	0.41 ± 0.03	98	Both conditions
Nemotron-Nano-12B-v2 (4 Mamba-2)	Word	90 / 86 / 93	0.01 ± 0.01	0.00 ± 0.01	0.01 ± 0.01	0.00 ± 0.01	27	Not detected
	Short word	90 / 92 / 92	0.00 ± 0.01	0.00 ± 0.01	0.01 ± 0.01	0.00 ± 0.01	32	Not detected
Nemotron-H-47B (5 Mamba-2)	Word	96 / 94 / 96	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	27	Not detected
	Short word	97 / 98 / 97	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	31	Not detected
Granite-4.0-H-Small (4 Mamba-2)	Word	83 / 78 / 81	0.24 ± 0.03	0.39 ± 0.03	0.48 ± 0.04	0.51 ± 0.03	99	Both conditions
	Paragraph	76 / 72 / 75	0.21 ± 0.03	0.35 ± 0.03	0.43 ± 0.03	0.46 ± 0.03	96	Both conditions
	Short word	87 / 82 / 87	0.25 ± 0.03	0.44 ± 0.03	0.47 ± 0.04	0.54 ± 0.02	100	Both conditions
	Numeric twin	96 / 95 / 94	0.07 ± 0.01	0.06 ± 0.01	0.13 ± 0.01	0.07 ± 0.00	88	Both conditions
	Numeric, one token	96 / 96 / 94	0.09 ± 0.01	0.05 ± 0.00	0.19 ± 0.01	0.07 ± 0.00	80	Both conditions
	Numeric, two tokens	94 / 89 / 93	$0.01 \pm 0.01^{\dagger}$	0.02 ± 0.01	$0.04 \pm 0.01^{\dagger}$	0.02 ± 0.01	62	Ordinary
RecurrentGemma-9B (2 RG-LRU)	Short word	70 / 68 / 68	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	40	Not detected
	Short word, present tense	84 / 84 / 84	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	46	Not detected

Table 29: Every matched-memory swap that was run. All recurrent memories after the last attention layer are replaced; 200 fresh identities per row; accuracy is unmodified seven-candidate top-1 for the source, recipient, and memory prompts (S / R / M). Scores are σ_{own} and σ_{test} (Equation 6) in logits; e^{σ} is the geometric-mean probability-ratio factor of Figure 5. "Consistent identities" is the share of identities whose ordinary σ_{own} and σ_{test} are both positive. "Read" applies one rule to every row: both scores have a 95% lower bound above zero in that condition. The two conditions here are ordinary execution and the shifted state, which is the pair this panel was preregistered under; Section 5 reports the skipped state as its second condition, and the two meet in the ordinary column. †The shifted state failed its validity check, so that condition is not scored. Task versions: "Story" uses a second sentence template, "Paragraph" writes the facts as one paragraph, "Short word" is the word task without its three distractor facts, and "present tense" states its facts in the present tense. Checkpoints and revisions are in Table 6; RecurrentGemma-9B runs in float32 for this experiment.

Further hybrids and the numeric twin. Nemotron-3-Nano-30B (four post-attention memories after attention L42), Nemotron-Nano-9B-v2 (eight), and Granite-4.0-H-Small (four, after attention L35) repeat the design on fresh identities with every site, endpoint, and rule fixed before the run (Table 29). The models where the swap finds nothing pass the same exact checks as the others, so the flat result is a measurement and not a failed install: in Nemotron-H-47B and Nano-12B-v2 the null, self-swap, installed-state, effective-input and stored-state checks are all exactly zero, and the single-token decode agrees with the full forward on 1,797 and 1,798 of the 1,800 comparisons. The carried-answer control is zero in Nemotron-H, Nano, and Granite-Small and small and positive in Nano-9B-v2 (0.03 in ordinary execution on the word task); both scores are measured net of it. Granite-Small's bf16 weights are widened to float32 as each module runs, so that the single-token decode matches the full forward. On Nemotron-H's numeric twin, solved as reliably as the word task, the swap leaves the logits unchanged. The twin differs from the word task in two ways at once: its values are numbers, and each two-digit room number takes three tokens where every one of the twenty-one place words takes one, so the answer is written across three positions rather than one. Separating the two calls for a value that is numeric and occupies one position.

Three versions do that. Writing rooms as `#4` makes the value one token after a shared `#`, so the answer is read at the position the swap acts on, as it is on the word task. Naming rooms with number words keeps one token per value but makes the value a word again, so it bounds the two accounts rather than separating them. The third holds the model fixed instead of the value: Granite-Small's tokenizer groups digits in threes, so in that one model `#40` is one token after the shared prefix and `#4320` is two, with the template, the value type and the model unchanged between them. That pair is the contrast the word-versus-number twin cannot make, since it varies the value's type and its length together.

All three ran, each on a fresh pool with its sites, gates and exact checks unchanged, and Granite-Small's own numeric twin ran beside its pair (Table 29). Length is not what the twin was measuring. In Nemotron-H the one-token numeric value leaves the read where the three-token one left it, undetectable beside the word task. Granite-Small, which reads its numeric twin weakly rather than not at all, gains as much from `#40` as from that twin and less from `#4320`, so within one model and

one template the read does not follow how many positions the answer occupies. Number words, one token and a word, move further than either numeric version in both models and still fall well short of places. What a value must be for these layers to read it is therefore open, and it is not the pair of differences the word-versus-number twin puts in play.

The cell we cannot fill is a place word written across several tokens. The assay requires the candidates of an identity to share a token prefix and differ at one position after it, which two-digit room numbers satisfy and multi-token place words do not: the place words that split in these tokenizers differ in their first token, not their last, and no family of ten shares a first token.

A third incoming state. Section 5 runs the swap under ordinary execution and under the skipped state. A third state removes the test prompt's lead entirely: the final token's residual entering S is taken from another prompt, so nothing the test prompt computed at that token reaches these layers. What remains of the test prompt in S is its convolution window and its gates; the memory prompt supplies the stored state alone. Both runs of the contrast see the same foreign residual, so whatever that residual carries cancels between the transposed memory and the control, and only the two exchanged bindings differ. A preregistered validity condition checks that the state still performs a key-based lookup rather than producing noise: with the foreign residual installed and no memory swap, the donor's key still raises its own place over a candidate that neither account predicts, by 0.37 $\pm$ 0.06 logits on the word task and 0.49 $\pm$ 0.07 on its paragraph version in Nemotron-H. The read is weaker under this state than under ordinary execution, which is why Section 5 does not use it.

Why the ordinary read exceeds the shifted read. Table 30 compares the three incoming states under the same memory swap: *ordinary* execution, the *skipped* state, which is the test prompt's own residual from before the last attention layer, and the *shifted* state, which is another prompt's. Ordinary minus skipped isolates that layer's final-token update; skipped minus shifted isolates where the residual comes from. In all four models most of the gap comes from the residual's source, not from the attention update.

	Increase in supplied location's score relative to	
Incoming state after the last attention layer	Memory's own answer	Recipient's answer
Nemotron-H: both factors contribute		
Ordinary execution	0.735 $\pm$ 0.048	1.006 $\pm$ 0.057
Own pre-attention state, attention skipped	0.678 $\pm$ 0.046	1.241 $\pm$ 0.075
Foreign prompt's pre-attention state	0.365 $\pm$ 0.045	0.740 $\pm$ 0.080
Ordinary minus own state (attention ran)	+0.058 $\pm$ 0.014	−0.236 $\pm$ 0.030
Own minus foreign state (foreign token)	+0.313 $\pm$ 0.050	+0.501 $\pm$ 0.091
Ordinary minus foreign state (total gap)	+0.370 $\pm$ 0.051	+0.266 $\pm$ 0.081
Nemotron-3 Nano: the inserted token is the cause		
Ordinary execution	0.177 $\pm$ 0.026	0.242 $\pm$ 0.031
Own pre-attention state, attention skipped	0.194 $\pm$ 0.030	0.335 $\pm$ 0.038
Foreign prompt's pre-attention state	0.115 $\pm$ 0.025	0.244 $\pm$ 0.036
Ordinary minus own state (attention ran)	−0.017 $\pm$ 0.009	−0.093 $\pm$ 0.013
Own minus foreign state (foreign token)	+0.079 $\pm$ 0.015	+0.091 $\pm$ 0.024
Ordinary minus foreign state (total gap)	+0.061 $\pm$ 0.015	−0.002 $\pm$ 0.025
Nemotron-Nano-9B-v2: the inserted token is the cause		
Ordinary execution	0.341 $\pm$ 0.030	0.413 $\pm$ 0.033
Own pre-attention state, attention skipped	0.397 $\pm$ 0.030	0.605 $\pm$ 0.038
Foreign prompt's pre-attention state	0.189 $\pm$ 0.026	0.377 $\pm$ 0.044
Ordinary minus own state (attention ran)	−0.056 $\pm$ 0.014	−0.192 $\pm$ 0.019
Own minus foreign state (foreign token)	+0.208 $\pm$ 0.027	+0.229 $\pm$ 0.046
Ordinary minus foreign state (total gap)	+0.152 $\pm$ 0.027	+0.037 $\pm$ 0.043
Granite-4.0-H-Small: the inserted token is the cause		
Ordinary execution	0.363 $\pm$ 0.028	0.474 $\pm$ 0.029
Own pre-attention state, attention skipped	0.409 $\pm$ 0.032	0.700 $\pm$ 0.039
Foreign prompt's pre-attention state	0.240 $\pm$ 0.030	0.484 $\pm$ 0.046
Ordinary minus own state (attention ran)	−0.046 $\pm$ 0.009	−0.226 $\pm$ 0.017
Own minus foreign state (foreign token)	+0.169 $\pm$ 0.023	+0.215 $\pm$ 0.040
Ordinary minus foreign state (total gap)	+0.123 $\pm$ 0.024	−0.010 $\pm$ 0.040

Table 30: Bypass-source experiment: transposed-minus-control change in the supplied location's score, relative to the memory prompt's own answer and to the recipient's answer, under three incoming states after the last attention layer, with the paired differences between them; 95% identity-bootstrap intervals over 200 identities per model.

F.4 ORDINARY ACCURACY UNDER RECURRENT-OUTPUT DISRUPTION

Condition (final token only)	Baseline (%)	Intervened (%)	Paired drop (points, 95%)	Correct→wrong	Wrong→correct
Word wording: source 70.7%, recipient 62.0%, 100 identities					
Post-attention outputs, calibration means (primary)	69.8	67.3	2.5 ± 1.9	9.0	12.6
Earlier five recurrent outputs, calibration means	69.8	62.2	7.6 ± 2.8	17.9	16.2
Post-attention outputs, zeros	69.8	50.7	19.1 ± 3.5	34.1	15.6
Earlier five recurrent outputs, zeros	69.8	32.3	37.5 ± 4.0	61.0	16.9
Post-attention minus earlier, means			-5.1 ± 3.1		
Retrospective: memory swap, transposed minus control memory, ordinary execution, all five memories					
Word wording (200 identities)	69.2		-2.8 ± 1.5	Control minus unswapped -0.2 ± 0.5	
Story wording (200 identities)	63.3		-3.6 ± 1.4	Control minus unswapped 0.0 ± 0.3	

Table 31: Ordinary seven-candidate top-1 accuracy when final-token Mamba outputs are replaced, Nemotron-H, no residual transplant. Paired drops are baseline minus intervened accuracy, averaged within identity (nine source prompts and one recipient prompt, weighted 9:1); transition rates are conditional on baseline correctness. The memory-swap rows compare the transposed with the control memory on the same units.

Does the ordinary answer depend on the post-attention outputs? A preregistered test runs the ordinary two-hop prompts with no transplant and replaces only the final-token outputs of the five post-attention Mamba layers with per-layer means from a separate calibration pool; the null, identity, installed-output, and prefix-scope checks are exact. The replacement lowers accuracy modestly, the memory swap has an accuracy consequence of the same order, and the same replacement at an earlier layer costs 7.6 points against these layers' 2.5, a paired difference of -5.1 ± 3.1 (Table 31). This is what the ordinary forward pass predicts: the answer is already resolved before these layers, so removing them costs little, and it is the direct evidence for that reading, which Section 3.3 reaches from the other side by measuring that attention carries the conversion. What Section 5 measures is not their necessity in an ordinary pass but what they do with what is in their memory, which is why the read appears as a change of odds in ordinary execution and as a change of the first-ranked answer once the incoming state carries no resolved answer of its own (Figure 5, right).

G TASK VARIANTS AND ROBUSTNESS

Task-family variants. Table 32 repeats the binding-task sweep under four variants with the same seeds and candidates: entity-less filler sentences after every fact, single-token color values in place of numbers, a one-hop entity–attribute binding with no intermediate key, and a bAbI-style two-fact task with typed facts and location answers. Every hybrid run keeps the key-dominant class at the last attention layer, the concentration stays high under filler sentences and color values, and on every two-hop variant the dense models stay below the hybrids. On the one-hop task, with no intermediate key to assemble, the rise splits over the last two attention layers; on the word task part of it falls on the Mamba layers after Nemotron-H's last attention layer (Appendix F).

Variant	Model	Gate (%)	Site	κ	π	Class	C
Filler context	Nemotron-H-8B	65/66	pre-L40	0.81 ± 0.12	0.36 ± 0.09	Key-dominant (dual)	0.90 ± 0.02
Filler context	Qwen2.5-7B	66/66	pre-L22	0.53 ± 0.13	0.53 ± 0.13	Balanced	0.52 ± 0.02
Filler context	Llama-3.1-8B	83/86	pre-L16	0.78 ± 0.10	0.37 ± 0.08	Key-dominant (dual)	0.35 ± 0.01
Word-valued attributes	Nemotron-H-8B	70/77	pre-L40	0.82 ± 0.12	0.20 ± 0.08	Key-dominant (dual)	0.89 ± 0.01
Word-valued attributes	Granite-H	66/62	pre-L35	0.66 ± 0.09	0.10 ± 0.06	Key-dominant (dual)	0.89 ± 0.04
Word-valued attributes	Qwen2.5-7B	66/68	pre-L21	0.39 ± 0.10	0.33 ± 0.08	Balanced	0.32 ± 0.01
Word-valued attributes	Llama-3.1-8B	85/80	pre-L16	0.68 ± 0.07	0.32 ± 0.06	Key-dominant (dual)	0.26 ± 0.01
One-hop binding	Nemotron-H-8B	100/100	pre-L40	3.17 ± 0.24	1.39 ± 0.21	Key-dominant (dual)	0.50 ± 0.01
One-hop binding	Granite-H	100/100	pre-L35	2.85 ± 0.18	0.41 ± 0.10	Key-dominant (dual)	0.57 ± 0.01
One-hop binding	Qwen2.5-7B	100/100	pre-L22	3.70 ± 0.39	3.75 ± 0.39	Dual, unbalanced	0.61 ± 0.01
One-hop binding	Llama-3.1-8B	100/100	pre-L16	3.39 ± 0.26	1.87 ± 0.22	Key-dominant (dual)	0.33 ± 0.01
bAbI-style two-fact	Nemotron-H-8B	70/72	pre-L40	1.75 ± 0.19	0.58 ± 0.14	Key-dominant (dual)	0.66 ± 0.02
bAbI-style two-fact	Qwen2.5-7B	77/77	pre-L22	1.80 ± 0.23	1.36 ± 0.21	Key-dominant (dual)	0.32 ± 0.01
bAbI-style two-fact	Llama-3.1-8B	70/62	pre-L17	1.19 ± 0.14	0.41 ± 0.12	Key-dominant (dual)	0.37 ± 0.02
bAbI-style, numeric values	Nemotron-H-8B	90/86	pre-L40	2.60 ± 0.19	0.78 ± 0.16	Key-dominant (dual)	0.83 ± 0.01

Table 32: Task-family variants of the binding task. Gate is source/recipient seven-way top-1; the site is the last attention layer for the sequential hybrids and the key peak otherwise; κ and π carry 97.5% intervals and C a 95% interval. Runs below the accuracy gate are absent.

H STATISTICAL PROCEDURES, THRESHOLDS, AND MULTIPLICITY

Preregistration and statistics. For every confirmatory experiment the construction, site, sample, and numerical thresholds were written to a configuration file before the held-out sample was run; we call such thresholds *prespecified*. We write $e \pm h$ for an estimate e whose interval has half-width h; every decision uses the interval's own bounds, which are close to symmetric. Intervals are 95% percentile bootstrap intervals over 10,000 replicates clustered by identity, except that the factorial classification uses 97.5% intervals for its two contrasts and a 90% interval for the equivalence test. Head-set and path comparisons use fixed random banks with add-one randomization or Holm correction. Every run passes exact checks: transplanting a recipient's own state or a zero patch changes no logit, and $R_{\text{by}} + P_{\text{all}} = F$ holds exactly.

H.1 THRESHOLD SENSITIVITY AND MULTIPLICITY

A site is *dual-supported* when the 97.5% lower bounds of both κ and π exceed zero; *key-dominant* when the 97.5% lower bounds of κ and of $\kappa - \pi$ exceed zero (with dual support when π's does as well), and position-dominant in the mirror case. It is *balanced* when it is dual-supported and the 90% interval of $\kappa - \pi$ lies inside ± 0.30 logits. Every threshold was fixed before the confirmatory factorial runs.

The rule of Section 2 has three fixed constants: the lower-bound level for κ, π, and $\kappa - \pi$, the interval level for the balance test, and the balanced band. Table 33 re-classifies every reference site under 18 combinations of these constants and counts the layers of each profile whose class changes. Every key-dominant site keeps its class under every setting; the only changes are two balanced sites at the edge of the band (Qwen2.5-7B pre-L22 and Falcon-H1-7B pre-L29), and few layers of any profile change class.

Model	Task	Site	Preregistered class	Classes across the grid	Layers changed (max / mean)
Nemotron-H-8B	Arrow	pre-L40	Key-dominant (dual)	Key-dominant (dual)	4 / 1.3 of 52
Nemotron-H-8B (pre-L39)	Arrow	pre-L39	Key-dominant (dual)	Key-dominant (dual)	4 / 1.3 of 52
Nemotron-H-8B	Colon	pre-L40	Key-dominant (dual)	Key-dominant (dual)	0 / 0.0 of 2
Nemotron-H-8B	Binding	pre-L40	Key-dominant (dual)	Key-dominant (dual)	0 / 0.0 of 52
Nemotron-H-4B	Arrow	pre-L40	Key-dominant (dual)	Key-dominant (dual)	0 / 0.0 of 52
Nemotron-H-4B	Binding	pre-L40	Key-dominant (dual)	Key-dominant (dual)	4 / 1.3 of 52
Granite-H	Arrow	pre-L35	Key-dominant (dual)	Key-dominant (dual)	3 / 1.7 of 40
Granite-H	Binding	pre-L35	Key-dominant (dual)	Key-dominant (dual)	1 / 0.7 of 40
Qwen2.5-7B	Arrow	pre-L22	Balanced	Balanced, dual, unbalanced	4 / 1.7 of 28
Qwen2.5-7B (pre-L26)	Arrow	pre-L26	Unresolved	Unresolved	5 / 2.2 of 28
Qwen2.5-7B	Binding	pre-L22	Balanced	Balanced	1 / 0.7 of 28
Qwen2.5-14B	Arrow	pre-L36	Position-dominant (dual)	Position-dominant (dual)	3 / 1.0 of 48
Qwen2.5-14B	Binding	pre-L36	Balanced	Balanced	2 / 1.0 of 48
Mistral-7B	Arrow	pre-L19	Key-dominant (dual)	Key-dominant (dual)	2 / 1.3 of 32
Mistral-7B	Binding	pre-L19	Key-dominant (dual)	Key-dominant (dual)	0 / 0.0 of 32
Llama-3.1-8B	Arrow	pre-L16	Key-dominant (dual)	Key-dominant (dual)	2 / 0.7 of 32
Llama-3.1-8B	Binding	pre-L16	Key-dominant (dual)	Key-dominant (dual)	2 / 0.7 of 32
Gemma-2-9B	Arrow	pre-L28	Key-dominant (dual)	Key-dominant (dual)	1 / 0.3 of 42
Gemma-2-9B	Binding	pre-L28	Key-dominant (dual)	Key-dominant (dual)	2 / 1.0 of 42
Falcon-H1-3B	Arrow	pre-L21	Key-dominant (dual)	Key-dominant (dual)	2 / 0.7 of 32
Falcon-H1-3B	Binding	pre-L21	Key-dominant (dual)	Key-dominant (dual)	4 / 2.3 of 32
Falcon-H1-7B	Arrow	pre-L29	Balanced	Balanced, key-dominant (dual), dual, unbalanced	2 / 1.2 of 44
Falcon-H1-7B	Binding	pre-L29	Balanced	Balanced	3 / 1.7 of 44

Table 33: Threshold sensitivity of the classification. "Site" is the last attention layer for sequential hybrids and the key peak otherwise; "classes across the grid" lists every class the site receives over the 18 settings; "layers changed" is the maximum and mean number of layers in the depth profile whose class differs from the preregistered setting. Binding-task rows are absent for models that failed the binding gate.

The concentration against the number of attention layers. $C = \max_l \Delta_l^+ / \sum_l \Delta_l^+$, with $\Delta_l^+ = \max(\Delta_l, 0)$, divides the largest positive increment by the sum over all layers, so a model with few attention layers has fewer places for the conversion than a dense model with attention in every layer. Table 34 therefore restricts the statistic to attention-containing layers (C_{attn}), compares it with the uniform null over those layers, and gives the effective attention count e^H. Against that null the ordering reverses: the dense models are the concentrated ones, because they spread the rise over a small part of the attention they have while the hybrids use nearly all of theirs. The raw comparison depends strongly on how many attention layers are available; these are observational comparisons of different pretrained models, not a matched test of changing the architecture.

		Arrow task			Binding task		
Model	Attn. blocks	C_{attn}	$C_{\text{attn}}/(1/n)$	e^H	C_{attn}	$C_{\text{attn}}/(1/n)$	e^H
Nemotron-H-8B	4	0.71	2.9	1.9	0.72	2.9	1.9
Nemotron-H-4B	4	0.66	2.7	1.9	0.71	2.8	1.8
Granite-H	4	0.73	2.9	1.8	0.57	2.3	2.0
RecurrentGemma-9B	12	0.80	9.6	2.0	—	—	—
Nemotron-3-Nano-30B	6	0.78	4.7	1.7	0.75	4.5	1.9
Falcon-H1-3B	31	0.59	18.3	4.6	0.56	17.4	4.8
Falcon-H1-7B	43	0.70	29.9	3.2	0.73	31.3	3.0
Qwen2.5-7B	27	0.61	16.5	2.9	0.54	14.5	3.2
Qwen2.5-14B	47	0.38	18.0	5.5	0.36	16.8	5.9
Mistral-7B	31	0.44	13.8	3.4	0.43	13.4	4.0
Llama-3.1-8B	31	0.35	11.0	7.1	0.32	10.0	8.0
Gemma-2-9B	41	0.65	26.6	4.1	0.68	27.9	3.6

Table 34: Resolution concentration restricted to attention-containing layers on the donor-answer effect. n counts attention-containing layers with a measured increment (the final layer has none); $C_{\text{attn}}/(1/n)$ is the ratio to the uniform null over those layers; e^H is the effective attention count carrying the positive increments.

Multiplicity across layers. The depth profiles are descriptive. Under a Bonferroni correction of both interval levels across every swept layer, the sequential hybrids keep their key-dominant band through the last attention layer and lose at most an edge layer, while single-layer bands in dense and parallel models do not survive the correction.

I Alternative explanations, boundary conditions, and controls

Intervention checks. The shifted memory experiments show no excess final-layer tail atypicality relative to unmodified recipients in either wording, and every memory-swap run passes its decoder and accuracy checks (Appendix F.3). These diagnostics characterize the interventions without certifying that every inserted state is on-distribution.

Related studies locate retrieval and answer-selection mechanisms in dense and recurrent models (Lieberum et al., 2023; Wiegreffe et al., 2025; Sharma et al., 2024; Ensign & Garriga-Alonso, 2024; Wang et al., 2026; Zani et al., 2025; Wu et al., 2025). Our hybrid panel uses architectures that combine state space computation with attention (Gu & Dao, 2024; Dao & Gu, 2024; Waleffe et al., 2024; NVIDIA et al., 2025). We also run path patching (Goldowsky-Dill et al., 2023) and causal scrubbing (Chan et al., 2022) and, because interventions can create atypical states (Makelov et al., 2024; Grant et al., 2026), track the distance of intervened trajectories from natural activations.

Alternative	Evidence against it
Fixed donor answer	One donor tracks three recipient bindings on 81% of crosses; S_A scrub destroys the effect
Recipient keeps its own answer A_r	$\Delta\ell(B_j) - \Delta\ell(A_r) = 4.17$ logits
Query token, not key, is transferred	Recipient maps q_j away from B_j in 97% of cells; excluding the rest leaves κ unchanged
State is only a row pointer	The key beats the position by 0.77 logits with a reference baseline; S_P scrub destroys the effect
Any nearby site works	Pre-L39 whole residual carries three quarters of the effect, same class
Any eight heads carry it	Top-8 path exceeds all 255 random eight-head paths
Heads merely relay donor outputs	Donor-clamped heads give negative key selectivity; $F - D_{\text{attn}} > 0$
The site is special only because we nominated it	Full depth sweeps of twelve models with no outcome selection; the three frozen sites keep their class
The result depends on the prompt format	Second surface form keeps the class at pre-L39 and pre-L40
The state barely encodes the key	Ridge probe retrieves the key on 78% of donors at pre-L40 (3-way)
Suffix clamp only causes nonspecific damage	Nominated-pair drop exceeds comparison-pair drop; same-key cross-donor rescue exceeds different-key control (Table 26)
Post-attention layers only pass along an answer	Memory prompts with the same own answer but transposed bindings redirect the lookup in four hybrids, in ordinary execution as well; the carried-answer control is zero in Nemotron-H and Nano (Table 29)
Nano's later lookup has no identified supplier	L37 interruption, specificity, rescue, and increased late-head dependence (Table 19)

Table 35: Alternative accounts and the evidence against each.